



NetScaler BLX 14.1

Machine translated content

Disclaimer

このコンテンツの正式なバージョンは英語で提供されています。Cloud Software Group ドキュメントのコンテンツの一部は、お客様の利便性のみを目的として機械翻訳されています。Cloud Software Group は機械翻訳されたコンテンツを管理していないため、誤り、不正確な情報、不適切な用語が含まれる場合があります。英語の原文から他言語への翻訳について、精度、信頼性、適合性、正確性、またはお使用の Cloud Software Group 製品またはサービスと機械翻訳されたコンテンツとの整合性に関する保証、該当するライセンス契約書またはサービス利用規約、あるいは Cloud Software Group とのその他すべての契約に基づき提供される保証、および製品またはサービスのドキュメントとの一致に関する保証は、明示的か黙示的かを問わず、かかるドキュメントの機械翻訳された範囲には適用されないものとします。機械翻訳されたコンテンツの使用に起因する損害または問題について、Cloud Software Group は責任を負わないものとします。

Contents

NetScaler BLX について	3
一般的なアーキテクチャ	4
サポートされている NetScaler の機能	6
NetScaler BLX ライセンス	7
システム要件	8
NetScaler BLX を導入	12
Linux ホストに NetScaler BLX パッケージをダウンロードします	12
Debian ベースの Linux ホストに NetScaler BLX をインストールする	13
RPM ベースの Linux ホストに NetScaler BLX をインストールする	15
NetScaler BLX の設定	17
NetScaler BLX 構成ファイル	17
NetScaler BLX を専用モードで構成する	19
NetScaler BLX 用の圧縮コアダンプを構成する	22
NetScaler BLX マネージドホストの設定	23
DPDK をサポートしない専用モードで NetScaler BLX 用の nsdrvd ドライバーを構成する	26
NetScaler BLX を起動する	29
NetScaler BLX にアクセスして NetScaler 機能を設定	30
NetScaler BLX クラスターのセットアップ	34
アップグレードとダウングレード	35
NetScaler BLX を AWS に展開	37
スタンドアロンの NetScaler BLX インスタンスを AWS に展開します	42
NetScaler BLX と GSLB を AWS に展開	46
よくある質問	49

Troubleshooting	53
NetScaler BLX の制限と使用ガイドライン	53

NetScaler BLX について

May 1, 2024

NetScaler BLX は、NetScaler ソフトウェアフォームファクターの 1 つです。市販の市販サーバー (COTS) 上のベアメタル Linux 上でネイティブに実行できます。

NetScaler は、アプリケーション固有のトラフィック分析を行い、Web アプリケーションのレイヤー 4 レイヤー 7 (L4~L7) ネットワークトラフィックをインテリジェントに分散、最適化、保護するアプリケーション Delivery Controller です。たとえば、NetScaler は、長期間有効な TCP 接続ではなく、個々の HTTP リクエストに基づいて決定を負荷分散します。詳しくは、「[NetScaler について](#)」を参照してください。

NetScaler BLX と他のソフトウェアフォームファクターの違いを教えてください

BLX、CPX、VPX は NetScaler ソフトウェアフォームファクターです。

- NetScaler BLX は、Linux システムでネイティブに動作する軽量のソフトウェアパッケージです。BLX は、ハイパーバイザーやコンテナのオーバーヘッドなしでシンプルさを実現し、パフォーマンスを向上させます。BLX は、選択したハードウェア上で Linux プロセスとして実行されます。
- CPX は NetScaler のコンテナ化されたバージョンであり、コンテナで実行する必要があります。
- VPX は仮想アプライアンスであり、サーバーにインストールされているハイパーバイザー上で実行する必要があります。

NetScaler BLX を選ぶ理由

BLX を使用する利点は次のとおりです。

- クラウド対応 -BLX はクラウド上での運用をゼロデイサポートします。クラウド上でプロビジョニングされた Linux 仮想マシン上でソフトウェアアプリケーションとして実行されるため、クラウド上で実行するための認証は必要ありません。
- 管理が容易 -Linux オペレーティングシステムの標準ツールを使用して BLX を監視および管理できます。BLX を既存のオーケストレーション設定に簡単に接続することもできます。
- サードパーティツールのシームレスな統合 -Linux 環境でサポートされているオープンソースツールを BLX とシームレスに統合できます。統合ごとに個別のプラグインを開発する必要はありません。
- 他のアプリケーションの共存 -BLX はソフトウェアアプリケーションとして動作します。他の Linux アプリケーションも同じホストで実行できます。
- **DPDK** サポート -BLX はデータプレーン開発キット (DPDK) の統合をサポートし、パフォーマンスを向上させます。DPDK オープンソースライブラリを使用してパフォーマンスを向上させ、パケット処理における Linux カーネルのボトルネックを克服します。

一般的なアーキテクチャ

May 1, 2024

NetScaler BLX は NetScaler のソフトウェアフォームファクターであり、他のフォームファクターと同じ機能を提供します。Linux ホスト上でユーザースペースアプリケーションとして実行されます。

BLX は、パケットの受信/送信と NIC ポートの管理に Linux ドライバーを使用します。起動時に作成される仮想イーサネット (veths) インターフェイス `blx0` および `blx1` は、Linux ホストと BLX 間の通信に使用されます。たとえば、BLX は veths を使用して Linux ホスト上の `syslog` デーモンにログ情報を送信します。

BLX のネットワークモードは、Linux ホストの NIC ポートを、ホスト上で実行されている他の Linux アプリケーションと共有するかどうかを定義します。BLX は、次のいずれかのネットワークモードで動作するように設定できます。

- 専用モード - Linux ホストの NIC ポートは BLX 専用で、他の Linux アプリケーションとは共有されません。
- 共有モード - Linux ホストの NIC ポートは、他の Linux アプリケーションと共有されます。

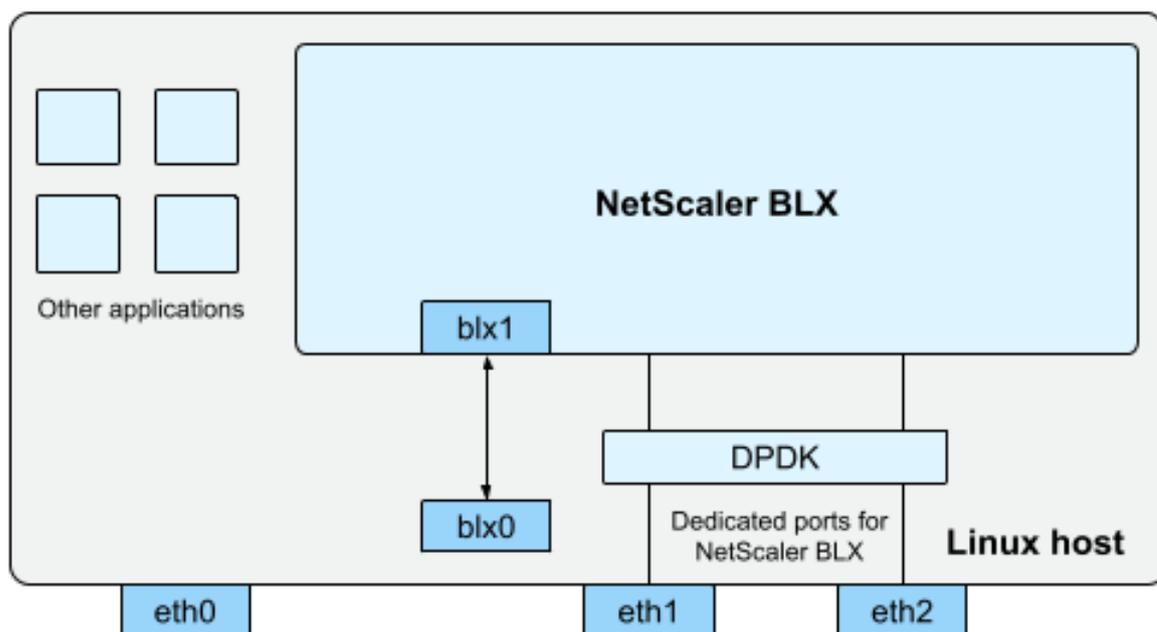
専用モードの **NetScaler BLX**

専用モードでは、`blx.conf` (BLX のブートアップ設定ファイル) で指定されている Linux ホストの NIC ポートは BLX 専用になります。これらの NIC ポートは、ホスト上で実行されている他のアプリケーションとは共有されません。BLX だけが専用の NIC ポートを見ることができます。

専用モードの IP アドレススキームは、従来の NetScaler と似ています。BLX には、NetScaler IP (NSIP)、仮想サーバー IP (VIP)、およびサブネット IP (SNIP) に異なる IP アドレスを割り当てることができます。

BLX は外部ネットワークからパケットを受信し、受信したパケットを処理し、設定された専用 Linux NIC ポートを介して直接応答します。Linux カーネルの TCP/IP スタックをバイパスして、パケットを処理するための本格的な TCP/IP スタックを備えています。BLX は Linux カーネルドライバーと直接やり取りして NIC ポートから未加工のパケットを取得します。

BLX は Linux カーネルのネットワークスタックをバイパスしますが、Linux カーネルメモリとユーザー空間メモリ間のパケット転送には依然としてオーバーヘッドがあります。このオーバーヘッドは、パケット処理の全体的なパフォーマンスに影響します。高いパケット処理性能を得るには、データプレーン開発キット (DPDK) 互換の NIC の使用をお勧めします。BLX がサポートする DPDK 互換 NIC のリストについては、[Linux ホストのハードウェア要件を参照してください](#)。



blx0 and blx1 - veth pair NIC ports created for communication between BLX and the Linux host
eth0, eth1, and eth2 - NIC ports available on the Linux host

DPDK は、ネットワークパフォーマンスを向上させるために使用されるオープンソースの Linux ライブラリとネットワークインターフェイスコントローラーのセットです。DPDK の詳細については、[DPDK の公式ウェブサイト https://www.dpdk.org/](https://www.dpdk.org/) を参照してください。

DPDK はカーネルメモリをバイパスし、パケットをユーザー空間メモリに直接配信して処理します。DPDK を Linux UIO モジュールと組み合わせると、BLX は、カーネルメモリからユーザー空間メモリにパケットをコピーするという Linux カーネルのオーバーヘッドを必要とせずにパケットを送受信できます。メモリが割り当てられると、DPDK はバッファを管理してパフォーマンスを向上させます。

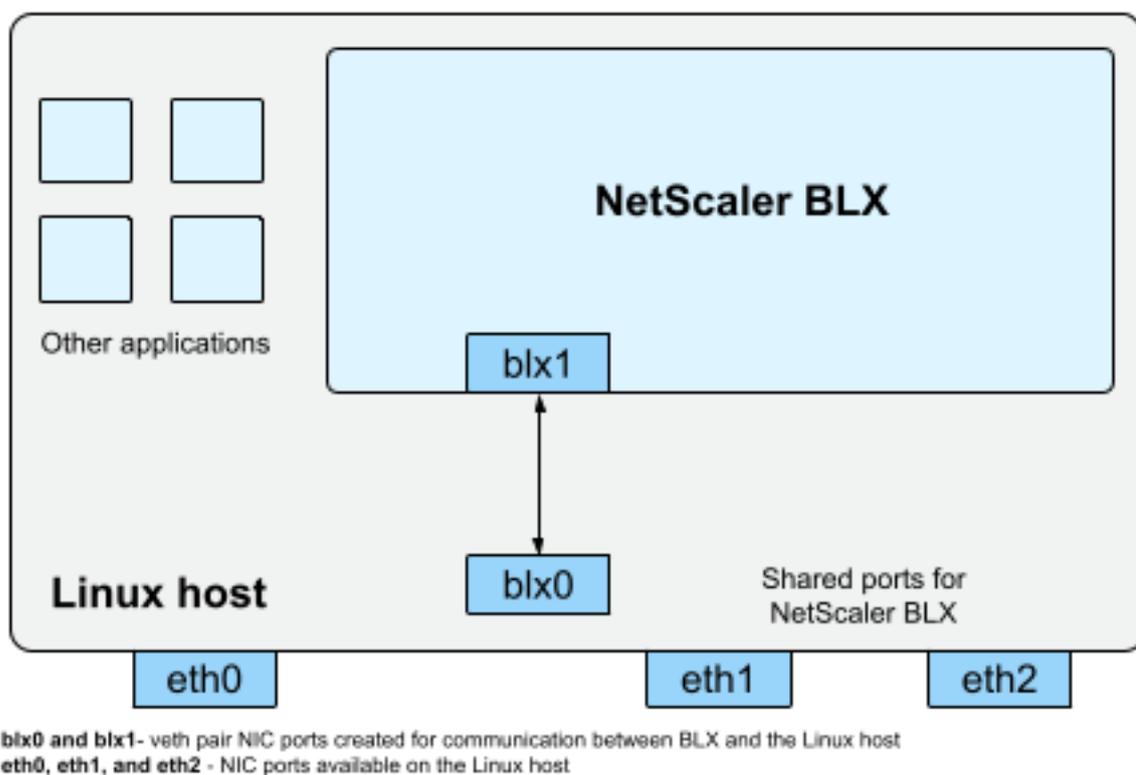
注:

以下の条件のいずれかが満たされると、BLX が DPDK サポートのない専用モードで起動することがあります。

- BLX は専用の DPDK 互換 NIC ポートをサポートしていません。
- DPDK は専用の NIC ポートをサポートしていません。

共有モードの **NetScaler BLX**

共有モードでは、Linux ホストの NIC ポートは他の Linux アプリケーションと共有され、パケットを送受信します。BLX には 192.0.0.1/24 の IP アドレスが自動的に割り当てられます。この IP アドレスは、管理およびデータトラフィックに使用されます。NetScaler が所有するすべての IP アドレス (NSIP、SNIP、VIP アドレスなど) の IP アドレスは 192.0.0.1 と同じですが、ポート番号は異なります。つまり、この単一の IP アドレス (192.0.0.1) は、NSIP、SNIP、および VIP として機能するために異なるポート番号を使用します。



ホストの Linux NIC ポートは BLX と他の Linux アプリケーション間で共有されるため、IP テーブルルールが NAT に追加されます。この IP テーブルルールは、ホストで受信したトラフィックを BLX に転送してさらに処理するために使用されます。

Linux ホストは、受信したパケットの宛先 IP アドレスを BLX の IP アドレス (192.0.0.1) に変換します。BLX は `blx0` および `blx1` 仮想インターフェイスを介してパケットを受信します。

BLX は受信したパケットを処理し、`blx1` および `blx0` 仮想インターフェイスを介して Linux カーネルに送信します。Linux ホストは BLX IP NAT テーブルを使用してこれらのパケットに NAT を実行し、Linux NIC ポートを介して宛先に送信します。

注:

実稼働環境では BLX を共有モードに設定することはお勧めしません。

サポートされている **NetScaler** の機能

September 30, 2024

NetScaler BLX は NetScaler のソフトウェア フォーム ファクターであり、他の NetScaler フォーム ファクターと同じ機能を提供します。

NetScaler の機能は、特定の要件に対応するために個別に、または組み合わせて構成できます。一部の機能は複数のカテゴリに該当しますが、多数の NetScaler 機能は一般に次のように分類できます。

- [アプリケーションスイッチングとトラフィック管理機能](#)
- [アプリケーション高速化機能](#)
- [アプリケーションセキュリティとファイアウォール機能](#)
- [アプリケーション可視性機能](#)
- [NetScaler Gateway アプリケーション](#)
- [スクリプト可能なモニターのサポート](#)

注:

- BLX 固有の機能制限については、[BLX の制限と使用ガイドライン](#)を参照してください。
- BLX でサポートされていない機能のリストについては、「[BLX でサポートされていない NetScaler 機能](#)」を参照してください。
- NetScaler リリース 14.1 ビルド 25.x 以降、NetScaler BLX は NetScaler Gateway をサポートします。詳細については、[NetScaler Gateway のドキュメント](#)を参照してください。

NetScaler BLX ライセンス

May 1, 2024

NetScaler は、組織の要件を満たす幅広い製品エディションとライセンスモデルを提供しています。

BLX は次の種類のライセンスをサポートしています。

- **エクスプレスライセンス** -デフォルトでは、BLX にはエクスプレスライセンスが付属しており、ライセンスファイルは必要ありません。詳しくは、「[NetScaler エクスプレスライセンス](#)」を参照してください。
- **固定帯域幅サブスクリプション** -固定帯域幅サブスクリプションは期間ベースのライセンスで、BLX が利用できる最大許容スループットを適用します。詳しくは、「[NetScaler 固定帯域幅ライセンス](#)」を参照してください。
- **NetScaler プールキャパシティライセンス** -プールキャパシティライセンスでは、NetScaler のさまざまなフォームファクター間で帯域幅またはインスタンスライセンスを共有できます。詳しくは、「[NetScaler プール容量](#)」を参照してください。

NetScaler BLX のホスト ID を取得

固定スループットライセンスと NetScaler プールキャパシティライセンスには、BLX のホスト ID が必要です。BLX のホスト ID を取得するには、NetScaler CLI で次のコマンドを実行します。

```
1 show hardware
```

サンプル出力:

```
> show hardware
Platform: ADC BLX 450091
Manufactured on: 11/9/23
CPU: 2199MHZ
Host Id: ██████████
Serial no: ██████████
Encoded serial no: ██████████
Netscaler UUID: ██████████
Done
> █
```

新しいライセンスのインストール

BLX にライセンスをインストールするには、自動または手動の方法を使用できます。BLX へのライセンスのインストールについて詳しくは、[NetScaler ライセンスガイド](#)を参照してください。

システム要件

September 30, 2024

NetScaler BLX を展開する前に、次の要件を確認してください。

- サポートされている Linux ディストリビューション
- Linux ホストのハードウェア要件
- オープンソースパッケージ

サポートされている **Linux** ディストリビューション

Linux ディストリビューション	NetScaler BLX 14.1	NetScaler BLX 13.1	NetScaler BLX 13.0
Red Hat Enterprise Linux (RHEL) 9.x	はい	はい	はい
RHEL8.x の場合	はい	はい	はい
RHEL 7.5 から 7.9	はい	はい	はい
8.x の場合	はい	はい	はい
CentOS 7.5 から 7.9	はい	はい	はい
Oracle Linux 8.x	はい	はい	はい

Linux ディストリビュー

ション	NetScaler BLX 14.1	NetScaler BLX 13.1	NetScaler BLX 13.0
Oracle Linux 7.5 から 7.9	はい	はい	はい
Ubuntu 22.04	はい (ビルド 12.30 以降)	いいえ	いいえ
Ubuntu 20.04	はい	はい	はい
Ubuntu 18.04	はい	はい	はい
Oracle Cloud Infrastructure (OCI) 上の Oracle Cloud Linux	はい	はい	はい
Amazon Linux 2	はい	はい	はい

注記:

- OCI では、DPDK を使用した BLX はサポートされていません。DPDK サポートなしで BLX を専用モードで実行できます。
- Amazon Linux では、DPDK を使用した BLX は Elastic Network Adapter (ENA) でのみサポートされます。

Linux ホストのハードウェア要件

| カテゴリ | DPDK をサポートする NetScaler BLX | ネットスケーラー BLX |

| ———— | ————— | ————— |

| ** プロセッサ ** | Intel または AMD x86-64 (64 ビット) プロセッサ | Intel または AMD x86-64 (64 ビット) プロセッサ |

| ** 最小 RAM ** | 2GB | 2GB |

| ** ドライバー ** | **Amazon EC2 Elastic Network Adapter ポート **: [ena](#) | Linux でサポートされているすべてのドライバー |

| ^^ | **Google Cloud Platform **: [gVNIC](#)、[virtIO](#) | ^^ |

| ^^ | **Intel ポート **: [igb](#)、[ixgbe](#)、および [i40e](#) | ^^ |

| ^^ | **KVM 準仮想化ドライバー **: [virtIO](#) | ^^ |

| ^^ | **Mellanox ConnectX-4 ポート **: [mlx5_core](#) | ^^ |

| ^^ | **Mellanox ConnectX-5 ポート **: [mlx5_core](#) | ^^ |

| ^^ | **Mellanox ConnectX-6 ポート **: [mlx5_core](#) | ^^ |

| ^^ | **VMware 仮想化プラットフォーム ネットワーク アダプタ ポート **: [vmxnet3](#) | ^^ |

注記:

- BLX は最大 9 個の NIC ポート (DPDK NIC ポート、非 DPDK NIC ポート、またはその両方の組み合わせ) をサポートします。
- Linux ホストは、DPDK をインストールするための最小システム要件を満たしている必要があります。DPDK の最小要件の詳細については、[公式 DPDK ドキュメント](#)を参照してください。

オープンソースパッケージ

Linux ホストには次の依存関係が必要です。

- 自動インストールされた依存関係
- 手動でインストールされた依存関係

自動インストールされた依存関係

BLX をインストールすると、標準のパブリック リポジトリから次の依存関係が Linux ホストに自動的にインストールされます。

RPM ベースの Linux ホスト	Debian ベースの Linux ホスト
<code>crontabs</code>	<code>build-essential:amd64</code>
<code>ethtool</code>	<code>coreutils</code>
<code>gcc</code>	<code>cpanminus:amd64</code>
<code>glibc(x86-32)</code> (バージョン 2.17 以降)	<code>cron:amd64</code>
<code>glibc(x86-64)</code> (バージョン 2.17-196 以降)	<code>ethtool</code>
<code>Requbsd</code>	<code>gcc:amd64</code>
<code>ibdb(x86-64)</code> (バージョン 5.3-21 以降)	<code>lib32gcc1:amd64</code> (バージョン 4.9 以降)
<code>libgcc(x86-32)</code> (バージョン 4.8.5 以降)	<code>lib32stdc++6:amd64</code> (バージョン 4.8.4 以降)
<code>libstdc++(x86-32)</code> (バージョン 4.8.5 以降)	<code>lib32z1:amd64</code> (バージョン 1.2.8 以降)
<code>libstdc++(x86-64)</code> (バージョン 4.8.5 以降)	<code>libbsd0</code> (バージョン 0.8.2 以降)
<code>make</code>	<code>libc6</code> (バージョン 2.17 以降)
<code>openssl-devel</code>	<code>libc6:amd64</code> (バージョン 2.19 以降)
<code>pciutils</code>	<code>libc6:i386</code> (バージョン 2.19 以降)
<code>perl</code>	<code>libc6-i386:amd64</code> (バージョン 2.19 以降)
<code>perl-App-cpanminus</code>	<code>libdb5.3:amd64</code> (バージョン 5.3.28 以降)

RPM ベースの Linux ホスト	Debian ベースの Linux ホスト
<code>perl-core</code>	<code>libhttp-message-perl:amd64</code>
<code>perl-CPAN</code>	<code>libio-socket-ssl-perl:amd64</code>
<code>perl-IO-Socket-SSL</code>	<code>liblwp-protocol-https-perl:amd64</code>
<code>perl-libwww-perl</code>	<code>libnuma1</code> (バージョン 2.0.11 以降)
<code>perl-LWP-Protocol-https</code>	<code>libssl-dev:amd64</code>
<code>perl-Sys-Syslog(x86-64)</code> (バージョン 0.33 以降)	<code>libstdc++6</code> (バージョン 4.8.5 以降)
<code>perl(x86-64)</code> (バージョン 5.16.3 以降)	<code>libswitch-perl:amd64</code>
<code>perl-XML-Writer</code>	<code>libwww-perl:amd64</code>
<code>procps-ng</code>	<code>libxml-parser-lite-perl:amd64</code>
<code>python3</code>	<code>libxml-writer-perl:amd64</code>
<code>rsyslog</code>	<code>perl:amd64</code> (バージョン 5.16.3 以降)
<code>sqlite-devel(x86-64)</code> (バージョン 3.7.17 以降)	<code>python3</code>
<code>systemd</code>	<code>rsyslog</code>
<code>tcpdump(x86-64)</code> (バージョン 4.9.2 以降)	<code>sqlite3:amd64</code> (バージョン 3.11.0 以降)
<code>zlib(x86-32)</code> (バージョン 1.2.7 以降)	<code>systemd</code>
	<code>tcpdump:amd64</code> (バージョン 4.9.2 以降)

手動でインストールされた依存関係

次の依存関係は、標準のパブリック リポジトリから Linux ホストに手動でインストールする必要があります。

- **RPM** ベースの **Linux** システムの場合は、**Extra Packages for Enterprise Linux (EPEL)** リポジトリをインストールします。

EPEL リポジトリのインストールについては、[EPEL](#)を参照してください。

- **Ubuntu** バージョン **18** 以降を実行している **Debian** ベースの **Linux** システムの場合は、**libc6:i386** リポジトリをインストールします。

パッケージをインストールするには、Linux シェルで次のコマンドを実行します。

- `dpkg --add-architecture i386`
- `apt update`
- `apt install libc6:i386`

- **AWS** 上の **Amazon Linux 2** の場合は、**EPEL** リポジトリをインストールします。

EPEL リポジトリをインストールするには、Amazon Linux 2 シェルで次の 3 つのコマンドを実行します。

1. `amazon-linux-extras install epel -y`
2. `yum-config-manager --enable epel`
3. `yum update`

EPEL リポジトリのインストールの詳細については、[AWS 公式ドキュメント](#)を参照してください。

NetScaler BLX を導入

August 30, 2024

Linux ホストに NetScaler BLX を導入する手順は、次のとおりです。

1. [Linux ホストに NetScaler BLX パッケージをダウンロードします。](#)
2. Linux ホストに NetScaler BLX をインストールします。
 - [NetScaler BLX を Debian ベースの Linux ホストにインストールします。](#)
 - [RPM ベースの Linux ホストに NetScaler BLX をインストールします。](#)

Linux ホストに NetScaler BLX パッケージをダウンロードします

May 1, 2024

NetScaler BLX インストールパッケージはダウンロードページでホストされています。インストールパッケージは、NetScaler 機能パッケージと DPDK サポート用パッケージで構成されています。

BLX インストールパッケージは TAR ファイルで、次のような命名規則があります。

- Debian ベースのパッケージの場合、`.blx-deb-<release number>-<build-number>.tar.gz`
- RPM ベースのパッケージの場合、`.blx-rpm-<release number>-<build-number>.tar.gz`

例:

`blx-deb-14.1-4.42.tar.gz`

`blx-rpm-14.1-4.42.tar.gz`

NetScaler BLX をダウンロード

1. Web ブラウザで[ダウンロードページ](#)を開きます。
2. ダウンロードページで、ダウンロードしたい **BLX** リリースを展開します。
3. BLX ビルドリンクをクリックします。
4. 「ファイルをダウンロード」をクリックして、BLX ビルドパッケージをダウンロードします。

注:

チェックサムは、ダウンロードしたビルドパッケージが Web サイトでホストされている実際のパッケージと一致していることを確認するために提供されます。チェックサムは、ビットが正しいことを確認するための重要なチェックです。

次のステップ

- [Debian ベースの Linux ホストに BLX をインストールします](#)
- [RPM ベースの Linux ホストに BLX をインストールします](#)

Debian ベースの Linux ホストに NetScaler BLX をインストールする

September 30, 2024

始める前に

- NetScaler BLX パッケージが Linux ホストで使用可能であることを確認します。BLX のダウンロード方法については、「[Linux ホストに BLX パッケージをダウンロードする](#)」を参照してください。
- 必要な依存関係をインストールするために、Linux ホストがインターネットにアクセスできることを確認します。自動インストールされる依存関係の詳細については、「[BLX の自動インストールされる依存関係](#)」を参照してください。
- Linux ホストに BLX をインストールするには、root または sudo 権限があることを確認してください。

NetScaler BLX をインストールする

1. BLX インストール パッケージを解凍し、作業ディレクトリを抽出した BLX インストール ディレクトリに変更します。

```
1 tar -xvzf blx-deb-<release number>-<build-number>.tar.gz
2
3 cd <path to the extracted BLX installation directory>
```

サンプル出力:

次のサンプル出力は、Linux ホストの `/var/blxinstall` ディレクトリにダウンロードされた BLX インストール パッケージ `blx-deb-14.1-4.42.tar.gz` が解凍されたことを示しています。次に、作業ディレクトリが抽出されたディレクトリ `blx-deb-14.1-4.42` に変更されます。

```
1 # cd /var/blxinstall
2
3 # tar -xvzf blx-deb-14.1-4.42.tar.gz
4
5 # cd blx-deb-14.1-4.42
6
7 # pwd
8 /var/blxinstall/blx-deb-14.1-4.42
```

2. BLX をインストールするには、次のコマンドを実行します。

```
1 apt install ./blx*.deb
```

注記:

Ubuntu バージョン 18 以降を実行している Debian ベースの Linux ホストでは、BLX のインストールが次の依存関係エラーで失敗する可能性があります。

```
The following packages have unmet dependencies: blx-core-libs:i386 : PreDepends: libc6:i386 (>= 2.19)but it is not installable
```

回避策: BLX をインストールする前に、Linux ホスト CLI で次のコマンドを実行します。

- `dpkg --add-architecture i386`
- `apt update`
- `apt install libc6:i386`

3. これはオプションの手順です。カスタム モニター機能を使用するには、`nsall.pl` および `nsntlm-lwp.pl` スクリプトを使用する前に、Comprehensive Perl Archive Network (CPAN) を使用してシステムに `perl(Authen::NTLM)` を手動でインストールする必要があります。

```
1 # cpan install Authen::NTLM
```

4. 次のコマンドを実行して、BLX のステータスを確認します。

```
1 systemctl status blx
```

デフォルトでは、BLX は非アクティブな状態です。

Linux ホストから **BLX** をアンインストールするには:

- Linux ホストで `apt remove blx` コマンドを実行して、BLX をアンインストールし、BLX 構成ファイルを保持します。
- Linux ホストで `apt purge blx` コマンドを実行して、BLX をアンインストールし、BLX 構成ファイルを削除します。

注記:

Linux ホストでは、BLX 関連のシステム ファイルの一部が削除されていないという警告メッセージが表示される場合があります。ただし、`apt purge blx` コマンドを実行すると、すべての BLX ファイルが削除されます。

次のステップ

- [BLX を構成する](#)

RPM ベースの **Linux** ホストに **NetScaler BLX** をインストールする

September 30, 2024

始める前に

- NetScaler BLX パッケージが Linux ホストで使用可能であることを確認します。BLX のダウンロード方法については、「[Linux ホストに BLX パッケージをダウンロードする](#)」を参照してください。
- 必要な依存関係をインストールするために、Linux ホストがインターネットにアクセスできることを確認します。自動インストールされる依存関係の詳細については、「[BLX の自動インストールされる依存関係](#)」を参照してください。
- Linux ホストに BLX をインストールするには、root または sudo 権限があることを確認してください。
- BLX バージョン 14.1 ビルド 17.x 以降では、Red Hat ベースの Linux ホストに BLX をインストールすると、Linux ホストで SELinux モジュールが使用可能な場合は SELinux ポリシーが適用されます。このポリシーにより、BLX を Linux ホスト上で実行できるようになります。SELinux ポリシーの詳細については、[SELinux ポリシー](#)を参照してください。

NetScaler BLX をインストールする

1. BLX インストールパッケージを解凍し、作業ディレクトリを抽出した BLX インストール ディレクトリに変更します。

```
1 tar -xvzf blx-rpm-<release number>-<build-number>.tar.gz
2
3 cd <path to the extracted BLX installation directory>
```

サンプル出力:

次のサンプル出力は、Linux ホストの `/var/blxinstall` ディレクトリにすでにダウンロードされている BLX インストール パッケージ `blx-rpm-14.1-4.42.tar.gz` が解凍されたことを示しています。次に、作業ディレクトリが抽出されたディレクトリ `blx-rpm-14.1-4.42` に変更されます。

```
1 # cd /var/blxinstall
2
3 # tar -xvzf blx-rpm-14.1-4.42.tar.gz
4
5 # cd blx-rpm-14.1-4.42
6
7 # pwd
8 /var/blxinstall/blx-rpm-14.1-4.42
```

2. Redhat ベースの Linux システムの場合は、Extra Packages for Enterprise Linux (EPEL) をインストールします。EPEL のインストール方法の詳細については、[EPEL ドキュメント](#)を参照してください。
3. これはオプションの手順です。カスタム モニター機能を使用するには、`nsall.pl` および `nsntlm-lwp.pl` スクリプトを使用する前に、Comprehensive Perl Archive Network (CPAN) を使用してシステムに `perl(Authen::NTLM)` を手動でインストールする必要があります。

```
1 # cpan install Authen::NTLM
```

4. BLX をインストールするには、次のコマンドを実行します。

```
1 yum install ./blx*.rpm
```

5. 次のコマンドを実行して、BLX のステータスを確認します。

```
1 systemctl status blx
```

デフォルトでは、BLX は非アクティブな状態です。

Linux ホストから **BLX** をアンインストールするには:

BLX をアンインストールするには、Linux ホストで `yum remove blx` コマンドを実行します。

次のステップ

- [BLX を構成する](#)

NetScaler BLX の設定

May 1, 2024

NetScaler BLX は、専用ネットワークモードまたは共有ネットワークモードで構成できます。BLX のネットワークモードは、Linux ホストの NIC ポートを、ホスト上で実行されている他の Linux アプリケーションと共有するかどうかを定義します。

- 専用ネットワークモード -BLX 専用の Linux ホストの NIC ポートは、Linux ホストで実行されている他のアプリケーションと共有されません。専用モードでの BLX の設定については、「[専用モードでの BLX の設定](#)」を参照してください。
- 共有ネットワークモード -Linux ホストの NIC ポートは、Linux ホストで実行されている他の Linux アプリケーションと共有されます。BLX を専用モードに設定しない場合、デフォルトでは BLX は共有モードで起動します。

注:

実稼働環境では BLX を共有モードに設定することはお勧めしません。

NetScaler BLX 構成ファイル

May 1, 2024

NetScaler BLX のインストールの一部として、構成ファイル (`blx.conf`) が Linux ホストに追加されます。`blx.conf` ファイルには、BLX の設定に使用できるさまざまなパラメータがあります。

デフォルトでは、BLX 設定ファイルではすべてのパラメーターにコメントが付きます（プレフィックスに # 記号が付きます）。

パラメータをコメント解除（プレフィックス # を削除）し、カスタム値に設定して特定のパラメータを有効にすることができます。

`blx.conf` ファイル内のパラメータを次の表に示します。

パラメーター	指定可能な値	デフォルト	説明
<code>worker-processes</code>	1 から 28	1	起動するワーカースタンププロセスの数。

パラメーター	指定可能な値	デフォルト	説明
<code>cpu-yield</code>	はい、有効、または 1	無効	CPU 利回りを設定します。CPU 生成を有効にすると、NSPPE は他のプロセスの CPU を回収します。
<code>core-dumps</code>	はい、有効、または 1	無効	BLX のコアダンプを設定します。
<code>syslog</code>	はい、有効、または 1	無効	シスログが Linux ホストのポート 514/UDP でリッスンできるようにします。BLX は Linux ホストのポート 514/UDP でリスニングしているシスログにログを送信します。
<code>ipaddress</code>	IP アドレス	BLX は Linux ホストに設定されているすべての IP アドレスでリッスンします	BLX の NSIP アドレスを専用モードで設定します。
<code>blx-managed-host</code>	1	無効	BLX 経由で Linux ホストへの SSH アクセスを設定します。
<code>host-ipaddress</code>	IP アドレス	なし	BLX 経由で Linux ホストに SSH アクセスしたい IP アドレスを設定します。 注: <code>blx-managed-host</code> パラメーターはパラメータと一緒に使用する必要があります。
<code>total-hugepage-mem</code>	最小:1G、最大:Linux ホストで使用可能	1G	BLX 用の DPDK ヒュージページメモリを設定します。
<code>interfaces</code>	Linux ホスト CLI に表示されている NIC ポート名	BLX はホストトラフィックをすべてのインターフェイスと共有します	Linux ホストの指定された NIC ポートは BLX 専用です。
<code>default</code>	IP アドレス	なし	専用インターフェースのデフォルトルートを設定します。
<code>cli-cmds</code>	NetScaler CLI コマンド	なし	BLX の起動時に実行したい NetScaler CLI コマンドを一覧表示します。

パラメーター	指定可能な値	デフォルト	説明
<code>nsdrv</code>	1、2、または 3	無効	パケット処理のパフォーマンスを向上させるように <code>nsdrv</code> ドライバーを設定します。

注:

パラメーターを無効にする場合は、`blx.conf` ファイルに #パラメータをコメント入力し、`systemctl restart blx` コマンドを使用して BLX を再起動します。変更は BLX の再起動後に適用されます。

NetScaler BLX を専用モードで構成する

September 30, 2024

専用モードでは、NetScaler BLX 専用の Linux ホストの NIC ポートは、Linux ホスト上の他のアプリケーションと共有されません。

高いパケット処理パフォーマンスを得るには、DPDK 対応 NIC を使用することをお勧めします。BLX でサポートされている DPDK 互換 NIC のリストについては、「[Linux ホストのハードウェア要件](#)」を参照してください。

`blx.conf` ファイルで次のパラメータを指定することにより、BLX を専用モードで設定できます。

- `worker-processes` - 開始するワーカースレッドの数。
- `interface` - BLX 専用の Linux ホストの NIC ポート。
- `ipaddress` - 専用モードの BLX の NSIP アドレス。
- `default` - BLX のデフォルト ルート。

`interface` パラメータで指定された NIC ポートが DPDK と互換性があり、BLX でサポートされている場合、NIC ポートは DPDK VFIO モジュールに自動的にバインドされます。NIC ポートが DPDK と互換性がない場合は、NIC ポートは非 DPDK ポートとして追加されます。BLX を起動すると、`interface` パラメータで指定されたすべてのポートが BLX の専用ポートとして追加されます。

注記:

BLX は一度に 1 種類の DPDK NIC ポートのみをサポートします。たとえば、すべて Mellanox ポート、またはすべて Intel ポートのいずれかです。

前提条件

- Linux ホストで IOMMU サポートが有効になっていることを確認します。IOMMU を有効にする方法については、Linux ホストのハードウェア ドキュメントを参照してください。
- BLX でサポートされている DPDK 互換 Mellanox ポートの場合は、Linux ホストに Mellanox OpenFabrics Enterprise Distribution (OFED) パッケージがインストールされていることを確認してください。Mellanox OFED パッケージのインストール方法については、[OFED ドキュメント](#)を参照してください。

NetScaler BLX を専用モードで構成する

BLX を専用モードで設定するには、Linux ホスト CLI を使用する必要があります。

1. 次のコマンドを実行して、`blx.conf` ファイルを開きます。

注記:

任意のテキスト エディターを使用して `blx.conf` ファイルを編集できます。

```
1 nano /etc/blx/blx.conf
```

2. `worker-processes` パラメータのコメントを解除し、BLX のパケット エンジンの数を指定します。

注記:

BLX でサポートされている VMXNET3 DPDK ポートの場合、ワーカー プロセスの数を 2 の累乗 (2^n) で指定する必要があります。たとえば、1、2、4、8 などです。

```
1 blx-system-config
2 {
3
4     ...
5     worker-processes: <number of worker processes>
6     ...
7 }
```

3. インターフェイス パラメータのコメントを解除し、BLX 専用にする Linux ホストの NIC ポートを指定します。

注:

- Linux ホスト CLI に表示されるポート名をスペースで区切って指定する必要があります。
- **AMD** プロセッサの場合、1 つ以上の IOMMU グループのすべての DPDK NIC ポートを指定する必要があります。IOMMU グループのすべての NIC ポートを指定しない場合は、その IOMMU グループの DPDK 互換 NIC ポートが非 DPDK 専用ポートとして BLX に追加されます。

```
1 blx-system-config
2 {
```

```

3
4     ...
5     interfaces: <interface1 interface2>
6     ...
7 }

```

4. `ipaddress` パラメータのコメントを解除し、BLX の NSIP アドレスを指定します。

```

1 blx-system-config
2 {
3
4     ...
5     ipaddress: <IP address>
6     ...
7 }

```

5. (オプション) `total-hugepage-mem` パラメータのコメントを解除し、DPDK Huge ページに割り当てるメモリを指定します。DPDK Huge ページの詳細については、[DPDK ドキュメント](#)を参照してください。

注記:

巨大ページの合計サイズは、メガバイト (MB または M) または ギガバイト (GB または G) で指定できます。たとえば、1024MB、1024M、1GB、1G などです。

```

1     blx-system-config
2     {
3
4         ...
5         total-hugepage-mem: <memory size>
6         ...
7     }

```

6. (オプション) デフォルトの `nsroot` ユーザーを非 `root` ユーザーに変更するには、`non-root-nsroot: yes` パラメータを追加します。BLX を非ルート `nsroot` モードで起動した後は、ルート `nsroot` ユーザーに戻したり、CLI 経由で `showtechsupport` コマンドを実行したりすることはできません。詳細については、[NetScaler BLX の制限事項と使用ガイドライン](#)を参照してください。

7. `default` パラメータのコメントを解除し、専用インターフェースのデフォルト ルートを指定します。

```

1 static-routes
2 {
3
4     ...
5     default <gateway IP address>
6     ...
7 }

```

8. `blx.conf` ファイルを保存します。

次のステップ

- [BLX を開始](#)

NetScaler BLX 用の圧縮コアダンプを構成する

May 1, 2024

`blx.conf`ファイル内の`core-dumps`パラメーターを使用して NetScaler BLX のコアダンプを有効にできます。

コアダンプは Linux ホスト上の`core_pattern`ファイル内のパターンに従って生成されます。

```
1 /proc/sys/kernel/core_pattern
```

`core_pattern`ファイルにパターンがない場合は、コアダンプ用に次のパターンがファイルに追加されます。

```
1 /var/core/core-%e-sig%s-user%u-group%g-pid%p-time%t
```

`blx.conf` ファイルを使用してコアダンプを有効にする

コアダンプを有効にするには、Linux ホスト CLI を使用する必要があります。

1. 次のコマンドを実行して`blx.conf`ファイルを開きます。

注:

`blx.conf` ファイルは、任意のテキストエディタを使用して編集できます。

```
1 nano /etc/blx/blx.conf
```

2. `core-dumps`パラメーターのコメントを解除して、1、`enable`または`yes`に設定します。

```
1 blx-system-config
2 {
3
4     ...
5     core-dumps: yes
6     ...
7 }
```

3. `blx.conf`ファイルを保存します。
4. BLX を再起動します。

```
1 systemctl restart blx
```

BLX が再起動すると、BLX のコアダンプが有効になります。

blx.conf ファイルを使用してコアダンプを無効にする

Linux ホスト CLI を使用してコアダンプを無効にする必要があります。

注:

Linux ホストでコアダンプを有効にすると、**core-dumps**パラメーターが**blx.conf**ファイル内でコメント (無効) になっていても、BLX のコアダンプが生成されます。

1. 次のコマンドを実行して**blx.conf**ファイルを開きます。

注:

blx.conf ファイルは、任意のテキストエディタを使用して編集できます。

```
1 nano /etc/blx/blx.conf
```

2. **core-dumps**パラメーターにコメントを付けます。

```
1 blx-system-config
2 {
3
4     ...
5     # core-dumps: yes
6     ...
7 }
```

3. **blx.conf**ファイルを保存します。
4. BLX を再起動します。

```
1 systemctl restart blx
```

BLX が再起動すると、BLX のコアダンプは無効になります。

NetScaler BLX マネージドホストの設定

August 30, 2024

NetScaler BLX のマネージドホスト機能を使用すると、BLX を介して Linux ホストを管理できます。この機能は、Linux ホストのすべての NIC ポートを専用ポートとして BLX に自動的に追加します。ポートが DPDK と互換性があり、BLX でサポートされている場合、それらは Linux ホスト上の DPDK VFIO モジュールにバインドされます。

BLX は、Linux ホストで最も優先順位の高いデフォルトルートの専用 NIC ポートの 1 つを選択します。選択したポートの IP アドレスとデフォルトルートが、BLX の NSIP アドレスとデフォルトルートとして追加されます。

Linux ホストの NIC ポートにデフォルトルートが設定されていない場合、BLX は IP アドレスが割り当てられた専用ポートをランダムに選択します。選択したポートの IP アドレスが BLX の NSIP アドレスとして追加されます。

デフォルトでは、Linux ホストへの SSH アクセスは NSIP アドレスのポート 9022 で有効になっています。

注:

- BLX は Linux ホストボンディングインターフェース (リンクアグリゲーションチャネル) を自動的に追加しませんが、ボンディングインターフェースのすべてのメンバーを BLX に追加します。
- デフォルトポートに複数の IP アドレスが割り当てられている場合、BLX は CLI にエラーメッセージを表示して、`blx.conf` ファイルに NSIP アドレスを手動で設定します。
- BLX を再起動すると、Linux ホストへのアクティブな SSH セッションはすべて閉じられます。接続を復元するには、ホストへの接続を再試行する必要があります。
- `blx.conf` ファイルに NSIP アドレスを手動で設定した場合、Linux ホストで使用可能なデフォルトルートは BLX に自動的に追加されません。
- `ns.conf` ファイル内の設定が `blx.conf` ファイルよりも優先されます。

Linux ホストへの SSH アクセスが可能な NetScaler BLX マネージドホストを有効にする

BLX 管理対象ホストを有効にするには、Linux ホスト CLI を使用する必要があります。

1. 次のコマンドを実行して `blx.conf` ファイルを開きます。

注:

`blx.conf` ファイルは、任意のテキストエディタを使用して編集できます。

```
1 nano /etc/blx/blx.conf
```

2. `blx-managed-host` パラメーターのコメントを解除して 1 に設定します。

```
1 blx-system-config
2 {
3
4     ...
5     blx-managed-host: 1
6     ...
7 }
```

3. 他のパラメーターが `blx.conf` ファイルにコメントされていることを確認してください。
4. `blx.conf` ファイルを保存します。
5. BLX を再起動します。

```
1 systemctl restart blx
```

BLX の再起動後、SSH クライアントを使用して Linux ホストと BLX に次の IP アドレスでアクセスできます。

- Linux ホスト- <NSIP address>:9022
- BLX- <NSIP address>:22

ポート 22 の Linux ホストへの SSH アクセスが必要な場合は、NSIP と Linux ホストに異なる IP アドレスを手動で設定できます。詳細については、「NSIP と Linux ホストに異なる IP アドレスを設定する」を参照してください。

NSIP と Linux ホストに異なる IP アドレスを設定します

前のセクションで説明した構成に加えて、`ipaddress`、`default` および `host-ipaddress` パラメータを使用して、NSIP と Linux ホスト（ホスト IP）に異なる IP アドレスを設定する必要があります。

Linux ホスト CLI を使用してポート 22 で SSH アクセスを有効にする必要があります。

1. 次のコマンドを実行して `blx.conf` ファイルを開きます。

注:

`blx.conf` ファイルは、任意のテキストエディタを使用して編集できます。

```
1 nano /etc/blx/blx.conf
```

2. `ipaddress` パラメーターのコメントを解除し、BLX にアクセスする NSIP アドレスを指定します。

```
1 blx-system-config
2 {
3
4     ...
5     ipaddress: <IP address>
6     ...
7 }
```

3. `host-ipaddress` パラメーターのコメントを解除し、Linux ホストにアクセスするホスト IP アドレスを指定します。

注:

IP アドレスは NSIP サブネット内にある必要があります。

```
1 blx-system-config
2 {
3
4     ...
5     host-ipaddress: <IP address>
6     ...
7 }
```

4. `default` パラメーターのコメントを解除し、デフォルトルートを設定します。

```
1 static-routes
2 {
```

```

3
4     ...
5     default <gateway IP address>
6     ...
7 }

```

5. `blx.conf` ファイルを保存します。

6. BLX を再起動します。

```
1 systemctl restart blx
```

7. BLX が再起動したら、NetScaler CLI で `show nsip` コマンドを実行してホスト IP アドレスを確認します。

```

> show nsip
-----
Ipaddress      Traffic Domain  Type           Mode  Arp  Icmp  Vserver  State
-----
1)             0                NetScaler IP   Active Enabled Enabled NA      Enabled
2)             0                SNIP           Active Enabled Enabled NA      Enabled
3)             0                Host IP       Active Enabled Enabled NA      Enabled
4)             0                SNIP           Passive Enabled Enabled NA      Enabled
5)             0                VIP            Passive Enabled Enabled Enabled Enabled
Done
>

```

SSH クライアントを使用して、次の IP アドレスの Linux ホストと BLX にアクセスできます。

- Linux ホスト- `<Host IP address>:22`
- BLX - `<NSIP address>:22`

DPDK をサポートしない専用モードで NetScaler BLX 用の `nsdrvd` ドライバーを構成する

May 1, 2024

NetScaler BLX は Linux カーネルのネットワークスタックをバイパスしますが、Linux カーネルメモリとユーザー空間メモリ間のパケット転送にはオーバーヘッドがあります。このオーバーヘッドは、パケット処理の全体的なパフォーマンスに影響します。

パケット処理性能を高めるには、DPDK 互換 NIC の使用をお勧めします。BLX がサポートする DPDK 互換 NIC のリストについては、[Linux ホストのハードウェア要件を参照してください](#)。

DPDK 互換 NIC がない場合は、`nsdrvd` ドライバーを使用して DPDK をサポートしなくてもパケット処理のパフォーマンスを向上させることができます。

Linux カーネルとのパケット送受信のやり取りはすべて `nsdrvd` ドライバーが行います。また、トラフィックを PE に分散します。`blx.conf` ファイル内の `nsdrvd` パラメーターを使用して `nsdrvd` ドライバーを構成できます。次の表は、`nsdrvd` パラメーターに指定できる値をまとめたものです。

可能な値	説明
<code>nsdrvd: 1</code>	専用ポートごとに1つのドライバープロセスが作成されます。RxとTxは順番に発生します。
<code>nsdrvd: 2</code>	専用ポートごとに1つのRxプロセスと1つのTxスレッドが作成されます。
<code>nsdrvd: 3</code>	専用ポートごとに1つのRxプロセスと2つのTxスレッドが作成されます。

`nsdrvd` ドライバーを使用するには、次の計算に基づいて、Linux ホストに少なくとも **n** 個のコアが必要です。

$$n \geq WP + (INT * P) + 1$$

各項目の意味は次のとおりです：

- **WP**-BLXのワーカースレッド(パケットエンジン)の数。`blx.conf`ファイル内の`worker-processes`パラメーターは、BLXのパケットエンジンの数を指定します。
- **INT**-BLX用の専用のLinuxホストNICポートの数。`blx.conf`ファイル内の`interface`パラメーターは、BLX専用のLinuxホストNICポートを指定します。
- **P**-BLXの`nsdrvd`ドライバースレッドの数。`blx.conf`ファイル内の`nsdrvd`パラメーターは、`nsdrvd`ドライバースレッドの数を指定します。

例: 次の構成のBLXには、少なくとも10個のコアが必要です。

- **WP** = 3 パケットエンジン
- **INT** = 2つの専用インターフェイス
- **P** = 3つの`nsdrvd`プロセス

$$n = WP + (INT * P) + 1 = (3 + 2 * 3 + 1) = 10$$

前提条件

- BLXが専用モードに設定されていることを確認します。設定手順については、[専用モードでのBLXの設定を参照してください](#)。
- BLXがサポートするDPDK互換NICに専用のNICポートが表示されていないことを確認してください。詳しくは、「[Linuxホストのハードウェア要件](#)」を参照してください。

`nsdrvd` ドライバーを有効にする

LinuxホストCLIを使用して`nsdrvd`ドライバーを有効にする必要があります。

1. 次のコマンドを実行して`blx.conf`ファイルを開きます。

注:

blx.conf ファイルは、任意のテキストエディタを使用して編集できます。

```
1 nano /etc/blx/blx.conf
```

2. nsdrvdパラメーターのコメントを解除し、値を 1、2、または 3 に設定します。

```
1 blx-system-config
2 {
3
4     ...
5     nsdrvd: <number of process>
6     ...
7 }
```

3. blx.conf ファイルを保存します。
4. BLX を再起動します。

```
1 systemctl restart blx
```

BLX が再起動すると、BLX で nsdrvd ドライバーが有効になります。

nsdrvd ドライバーを無効にする

Linux ホスト CLI を使用して nsdrvd ドライバーを無効にする必要があります。

1. nsdrvd ドライバーを無効にするには、blx.conf ファイル内の nsdrvd パラメーターにコメントを付けます。

```
1 blx-system-config
2 {
3
4     ...
5     # nsdrvd: 2
6     ...
7 }
```

2. BLX を再起動します。

```
1 systemctl restart blx
```

BLX が再起動すると、BLX の nsdrvd ドライバーは無効になります。

NetScaler BLX を起動する

August 30, 2024

NetScaler BLX は、Linux ホスト上で動作するアプリケーションです。BLX をインストールして設定したら、Linux ホスト CLI で次のコマンドを実行して BLX を起動する必要があります。

```
1 systemctl start blx
```

注:

- BLX の起動には最大 45 秒かかる場合があります。
- CentOS バージョン 8.x または Oracle Linux バージョン 8.x に展開された BLX バージョン 14.1 ビルド 12.35 以前では、Linux ホストでSELinuxポリシーが有効になっていると、起動しないか、正しく機能しないことがあります。

回避策: Linux ホストでSELinuxを無効にします。

1. `nano /etc/selinux/config` コマンドを使用して Linux ホスト上でSELinux構成ファイルを開きます。
2. `SELINUX=disabled` を設定してファイルを保存します。
3. `systemctl restart blx` コマンドを使用して BLX を再起動します。

NetScaler BLX のステータスを確認する

- Linux ホスト CLI で次のコマンドを実行すると、BLX のステータスを確認できます。

```
1 systemctl status blx
```

BLX のステータスは `active (exited)` でなければなりません。

```
[root@blx-rpm ~]# systemctl status blx
● blx.service - BLX service
   Loaded: loaded (/usr/lib/systemd/system/blx.service; enabled; vendor preset: disabled)
   Active: active (exited) since Thu 2023-12-14 06:10:53 UTC; 1 weeks 0 days ago
     Process: 4178485 ExecStart=/root/.blx/blx-pre-start.sh (code=exited, status=0/SUCCESS)
     Process: 4178483 ExecStartPre=/bin/bash -c ${CHCON} (code=exited, status=0/SUCCESS)
     Process: 4178138 ExecStartPre=/usr/sbin/blx-helper.sh (code=exited, status=0/SUCCESS)
    Main PID: 4178485 (code=exited, status=0/SUCCESS)
       Tasks: 0 (limit: 23565)
      Memory: 0B
     CGroup: /system.slice/blx.service
```

- Linux ホストで実行されている NetScaler プロセスを確認するには、Linux ホスト CLI で次のコマンドを実行します。

```
1 ps aux | grep ns
```

注:

`nsppe` プロセスが Linux ホストで実行されていることを確認します。

例: `root 68332 2.7 5.5 485264 442084 ? Ss 16:25 0:02 /usr/sbin/nsppe 1`

NetScaler BLX を停止

BLX と関連プロセスを停止するには、Linux ホスト CLI で次のコマンドを実行します。

```
1 systemctl stop blx
```

注:

Linux ホストを再起動すると、再起動前に停止していても、Linux ホストの再起動後に BLX が自動的に起動します。Linux ホストの再起動後に BLX が自動的に起動しないようにするには、`systemctl disable blx` コマンドを実行する必要があります。

次のステップ

- [NetScaler BLX にアクセスして NetScaler 機能を設定](#)

NetScaler BLX にアクセスして NetScaler 機能を設定

August 30, 2024

NetScaler BLX には、次のいずれかの方法でアクセスできます。

- NetScaler CLI
- NetScaler GUI
- NetScaler NITRO REST API

はじめに

- BLX が Linux ホスト上で稼働していることを確認します。BLX の起動方法の詳細については、「[BLX の起動](#)」を参照してください。

パスワードの要件

- デフォルトの **admin (nsroot)** パスワードで初めてログインすると、セキュリティ上の理由から BLX からパスワードを変更するように求められます。パスワードを変更したら、設定を保存する必要があります。設定が保存されず、BLX が再起動した場合は、デフォルトパスワードで再度ログインする必要があります。
- BLX では、すべてのローカルシステムユーザーに対して強力なパスワード強制がデフォルトで有効になっています。強力なパスワードのデフォルトの最小文字数は 4 文字です。強力なパスワードには以下が含まれている必要があります。
 - 1 つの小文字。
 - 1 文字の大文字。
 - 1 つの数字。
 - セット内の 1 つの特殊文字 (!、@、#、(、)、\$、%、^、&、および*)。

注:

BLX の各システムユーザーのパスワードが、厳密なパスワード基準を満たしていることを確認してください。

- 強力なパスワード基準について詳しくは、「[NetScaler でパスワードの複雑さを強制する方法](#)」を参照してください。

NetScaler BLX にアクセスし、NetScaler CLI を使用して NetScaler 機能を構成する

BLX には、NetScaler CLI コマンドを実行して BLX 上で NetScaler 機能を構成できるコマンドラインインターフェイス (CLI) があります。

ワークステーションからセキュアシェル (SSH) 経由で接続することで、BLX にリモートアクセスできます。

次の表に、SSH 経由で NetScaler ADC CLI を使用できる IP アドレスとポートの一覧を示します。

BLX 展開モード	SSH 経由で NetScaler ADC CLI にアクセスするための IP アドレスとポート
専用	<NetScaler IP address (NSIP)>:22
共有	<Linux host IP address>:9022

NetScaler CLI を使用して **NetScaler BLX** にアクセスするには:

1. ワークステーションから SSH クライアントを開きます。
2. BLX の CLI が使用可能な IP アドレスとポートを指定し、CLI に接続します。
3. BLX のログイン認証情報を使用して BLX にログインします。

NetScaler CLI コマンドについて詳しくは、『[NetScaler コマンドリファレンスガイド](#)』を参照してください。

NetScaler BLX にアクセスし、NetScaler GUI を使用して NetScaler 機能を構成する

NetScaler GUI には、構成ユーティリティとダッシュボードユーティリティが含まれています。

次の表に、NetScaler GUI が使用できるデフォルトの IP アドレスとポートを示します。

BLX 展開モード	アクセスタイプ	NetScaler GUI にアクセスするための IP アドレスとポート
専用	HTTP	<NetScaler IP address (NSIP)>:80
専用	HTTPS	<NetScaler IP address (NSIP)>:443
共有	HTTP	<Linux host IP address>:9080
共有	HTTPS	<Linux host IP address>:9443

これらのデフォルトポート番号は `blx.conf` ファイル内で変更できます。 `blx.conf` ファイルを変更したら、BLX を再起動する必要があります。

注:

- 共有モードでは、 `set ns param` コマンドを使用して HTTP と HTTPS のデフォルトの管理ポート番号を変更することはできません。
- 専用モードでは、 `set ns param` コマンドを使用して HTTP と HTTPS のデフォルトの管理ポート番号を変更できます。ただし、 `unset ns param` または `clear config full` コマンドを使用しても、デフォルトのポート番号は復元されません。

NetScaler GUI により、BLX ログイン資格情報の入力を求められます。GUI にログインすると、NetScaler GUI を使用して NetScaler 機能を構成できます。

GUI を使用して NetScaler BLX にアクセスする

1. Web ブラウザーを開きます。

2. 次のいずれかのアクセス方法を使用します。

- HTTP アクセスの場合は、[URL] フィールドに次のように入力します。 <NetScaler BLX IP address (NSIP)>:<HTTP port>
- HTTPS アクセスの場合は、[URL] フィールドに次のように入力します。 <NetScaler BLX IP address (NSIP)>:<HTTPS port>

3. ログインページで、**NetScaler BLX** のログイン資格情報を入力し、「ログイン」をクリックします。

NITRO API を使用して **NetScaler BLX** にアクセスし、**NetScaler** 機能を構成します

NetScaler NITRO API を使用して NetScaler 機能を構成できます。NITRO では、Representational State Transfer (REST) インターフェイスを介して機能が提供されます。そのため、NITRO アプリケーションはあらゆるプログラミング言語で開発することができます。また、Java、.NET、または Python で開発する必要があるアプリケーションの場合、NITRO API は、個別のソフトウェア開発キット (SDK) としてパッケージ化された関連ライブラリを通じて公開されます。

NetScaler GUI と同様に、NITRO API リクエストは BLX 管理 IP アドレスの HTTP ポートまたは HTTPS ポートに送信する必要があります。

専用モードで **NetScaler BLX** にアクセスする

- Web ブラウザで NITRO API を使用して専用モードで BLX を設定するには、次のように入力します。

```
http://<NetScaler BLX IP address (NSIP)>:<HTTP port>/nitro/v1/  
config/<resource-type>
```

```
https://<NetScaler BLX IP address (NSIP)>:<HTTPS port>/nitro/v1/  
config/<resource-type>
```

- Web ブラウザで NITRO API を使用して専用モードで BLX の統計情報を取得するには、次のように入力します。

```
http://<NetScaler BLX IP address (NSIP)>:<HTTP port>/nitro/v1/  
stats/<resource-type>
```

```
https://<NetScaler BLX IP address (NSIP)>:<HTTPS port>/nitro/v1/  
stats/<resource-type>
```

共有モードで **NetScaler BLX** にアクセスする

- Web ブラウザで NITRO API を使用して BLX を共有モードに設定するには、次のように入力します。

```
http://<Linux host IP address>:<HTTP port>/nitro/v1/config/<  
resource-type>
```

```
https://<Linux host IP address>:<HTTPS port>/nitro/v1/config/<  
resource-type>
```

- Web ブラウザで NITRO API を使用して共有モードの BLX の統計情報を取得するには、次のように入力します。

```
http://<Linux host IP address>:<HTTP port>/nitro/v1/stats/<resource-type>
```

```
https://<Linux host IP address>:<HTTPS port>/nitro/v1/stats/<resource-type>
```

NetScaler NITRO API の使用について詳しくは、「[NetScaler BLX NITRO API](#)」を参照してください。

次のステップ

- [NetScaler 機能の設定](#)

NetScaler BLX クラスターのセットアップ

May 1, 2024

NetScaler BLX クラスターは、1つのシステムとして連携して動作する BLX インスタンスのグループです。各 BLX インスタンスはノードと呼ばれます。BLX クラスターには、ノードとして1つのインスタンスまたは最大 32 のインスタンスを含めることができます。

はじめに

- NetScaler クラスター機能を理解していることを確認してください。詳しくは、「[NetScaler クラスター](#)」を参照してください。
- すべての BLX インスタンスの Linux ホストに次の設定が存在することを確認してください。
 - NTP は各 Linux ホストで設定されます。

注:

1 - Oracle Linux での NTP の設定については、Oracle Linux [のマニュアルを参照してください](<https://docs.oracle.com/en/operating-systems/oracle-linux/index.html>)。

- * [Ubuntu Linux](#) 上で NTP を設定する方法については、[Ubuntu Linux のドキュメント](#)を参照してください。
- * [CentOS Linux](#) で NTP を設定する方法については、[CentOS Linux のマニュアル](#)を参照してください。

- BLX ログのロギングと `rsyslog` 設定が構成されます。

- すべての BLX インスタンスでコアダンプが有効になっていることを確認します。コアダンプの有効化について詳しくは、「[BLX の圧縮コアダンプの設定](#)」を参照してください。
- クラスターは、専用モードに設定された BLX インスタンスでのみサポートされます。
- NetScaler クラスターの一般的な前提条件はすべて BLX クラスターに適用されます。

注:

NetScaler クラスターを設定するための一般的な前提条件について詳しくは、「[NetScaler クラスターの一般的な前提条件](#)」を参照してください。

- BLX クラスターでサポートされている NetScaler 機能について詳しくは、「[BLX クラスターの NetScaler 機能サポートマトリックス](#)」を参照してください。
- Terraform を使用して NetScaler 展開を自動化する方法については、以下を参照してください。
 - [GitHub の NetScaler テラフォームプロバイダー](#)
 - [GitHub の NetScaler テラフォーム自動化スクリプト](#)
- BLX クラスター設定はパブリッククラウドプラットフォームではサポートされていません。たとえば、AWS クラウド。

NetScaler BLX クラスターの制限事項

BLX クラスターには次の制限があります。

- INC モードはサポートされていません。
- CLAG ベースのトラフィック分散はサポートされていません。
- スタンドアロン BLX のすべての制限は BLX クラスターにも適用されます。
スタンドアロン BLX の制限について詳しくは、「[BLXの制限事項](#)」を参照してください。

NetScaler BLX クラスターのセットアップ

BLX クラスターをセットアップするには、[NetScaler クラスターで NetScaler クラスターをセットアップするための一般的な手順](#)に従います。

アップグレードとダウングレード

May 1, 2024

NetScaler BLX の各リリースでは、機能が強化された新機能と更新された機能が提供されます。BLX を最新リリースにアップグレードして、新機能やバグ修正を利用することをお勧めします。[すべてのリリース発表に付随するリリースノートには](#)、拡張機能、既知の問題点、およびバグ修正の包括的なリストが含まれています。

はじめに

- 組織のサポート契約を評価する必要があります。NetScaler テクニカルサポートまたは NetScaler 認定パートナーからのサポートについては、サポート契約と連絡先の詳細を文書化してください。
- アップグレードする前に、ライセンスフレームワークと使用できるライセンスの種類を理解しておくことも重要です。詳しくは、「[BLX ライセンス](#)」を参照してください。
- 「[新規および廃止されたコマンド、パラメータ、SNMP OID](#)」のトピックを確認する必要があります。
- BLX の設定ファイルをバックアップします。バックアップと復元の方法については、「[失われた構成を回復するために NetScaler をバックアップおよび復元する方法](#)」を参照してください。

注:

バックアップされるファイルのリストの詳細については、「[バックアップされたファイルのリスト](#)」を参照してください。

- BLX のアップグレードまたはダウングレードは、BLX のインストールと同じです。Linux ホストのパッケージマネージャーは、ビルド番号に基づいてアップグレードまたはダウングレード操作を管理します。
- 必要に応じて、テスト環境で BLX をアップグレードまたはダウングレードしてみてください。

NetScaler BLX のアップグレード

1. アップグレードしたい BLX リリースパッケージをダウンロードします。詳しくは、「[Linux ホストでの BLX パッケージのダウンロード](#)」を参照してください。
2. Linux ホスト CLI を使用してダウンロードしたパッケージをインストールします。詳細については、「[Debian ベースの Linux ホストに BLX をインストールする](#)」または「[RPM ベースの Linux ホストに BLX をインストールする](#)」を参照してください。

注:

BLX 管理対象ホストを設定した場合は、ダウンロードしたパッケージをインストールする前に次の手順を実行してください。

- ホスト IP アドレスにログインします。
- `systemctl stop blx` コマンドを使用して BLX を停止します。
- ホスト IP アドレスに再接続します。

NetScaler BLX のダウングレード

1. ダウングレードしたい BLX リリースパッケージをダウンロードします。詳しくは、「[Linux ホストでの BLX パッケージのダウンロード](#)」を参照してください。
2. Linux ホスト CLI を使用してダウンロードしたパッケージをインストールします。詳細については、「[Debian ベースの Linux ホストに BLX をインストールする](#)」または「[RPM ベースの Linux ホストに BLX をインストールする](#)」を参照してください。

注:

- BLX 管理対象ホストを設定した場合は、ダウンロードしたパッケージをインストールする前に次の手順を実行してください。
 - Log in to the host IP address.
 - Stop BLX using the `systemctl stop blx` command.
 - Reconnect to host IP address.

3. 自動設定機能を使用して設定された BLX 管理対象ホストの場合、BLX をバージョン 13.1 ビルド 45.64 以降からバージョン 13.1 ビルド 42.47 またはそれ以前のバージョンにダウングレードするときは、次の手順を実行します。
 - a) ディレクトリ/`etc/blx/`内の`blx.conf`ファイルを開きます。
 - b) `interfaces`パラメーターのコメントを解除し、`blx.conf`ファイル内の BLX 専用にしたがいインターフェースを指定します。
 - c) `ip-address`パラメーターのコメントを解除し、BLX の NSIP アドレスを指定します。
 - d) `default`パラメーターのコメントを解除し、BLX のデフォルトルート指定します。
 - e) `blx.conf`ファイルを保存します。
 - f) `systemctl restart blx`コマンドを使用して BLX を再起動します。

アップグレードまたはダウングレード後に **NetScaler BLX** でエンティティのステータスを確認する

BLX がアップグレードまたはダウングレードされたら、次の点を確認してください。

- 仮想サーバは UP 状態です
- モニタはアップ状態です
- すべての証明書は BLX にあります
- すべてのライセンスは BLX にあります

NetScaler BLX を AWS に展開

August 30, 2024

NetScaler BLX は、AWS で利用可能な Linux インスタンスに展開できます。BLX を AWS に展開すると、AWS のクラウドコンピューティング機能と NetScaler 機能をビジネスニーズに合わせて使用できます。

AWS の用語

このセクションでは、よく使用される AWS の用語と語句のリストについて説明します。詳細については、[AWS 用語集](#)を参照してください。

用語	定義
Amazon マシンイメージ (AMI)	マシンイメージ。クラウド内の仮想サーバーであるインスタンスを起動するのに必要な情報を提供します。
Elastic Block Store	AWS クラウドで Amazon EC2 インスタンスと一緒に使用される、永続ブロックストレージボリュームを提供します。
Simple Storage Service (S3)	Internet 用のストレージ。Web 規模のコンピューティングを開発者が簡単に実施できるように設計されています。
Elastic Compute Cloud (EC2)	安全でサイズ変更可能なコンピューティングキャパシティをクラウドで提供するウェブサービス。Web 規模のクラウドコンピューティングを開発者が簡単に実施できるように設計されています。
Elastic Load Balancing (ELB)	受信アプリケーショントラフィックを複数のアベイラビリティゾーンの複数の EC2 インスタンスに分散します。これによってアプリケーションのフォールトトレランスが増加します。
エラスティックネットワークインターフェイス (ENI)	仮想プライベートクラウド (VPC) のインスタンスにアタッチできる仮想ネットワークインターフェイス。
Elastic IP (EIP) アドレス	Amazon EC2 または Amazon VPC で割り当てられ、インスタンスにアタッチされた、静的パブリック IPv4 アドレスです。Elastic IP アドレスは特定のインスタンスではなく、お使いのアカウントに関連しています。ニーズの変化に応じて、割り当て、アタッチ、デタッチ、および解放が簡単にできるため、Elastic (融通が利く) と呼ばれています。

用語	定義
インスタンスタイプ	Amazon EC2 では、さまざまなユースケースに対応できるように最適化された幅広い種類のインスタンスを提供しています。インスタンスタイプを構成する CPU、メモリ、ストレージ、およびネットワーク機能の組み合わせはさまざまで、アプリケーションに合わせて最適なリソースの組み合わせを柔軟に選択できます。
Identity and Access Management (IAM)	AWS で ID が実行できること、または実行できないことを決定する許可ポリシーを持つ AWS の ID。IAM ロールを使うことで EC2 インスタンス上で実行されるアプリケーションが、AWS リソースに安全にアクセスできるようになります。
インターネットゲートウェイ	ネットワークをインターネットに接続します。VPC 外部の IP アドレスのトラフィックをインターネットゲートウェイにルーティングできます。
キーペア	身元を電子的に証明するために使用する一連の資格情報。キーペアはプライベートキーとパブリックキーで構成されます。
ルートテーブル	関連付けられているサブネットからのトラフィックを制御するための一連のルーティング規則。1 つのルートテーブルに対して複数のサブネットを関連付けることができますが、各サブネットは一度に 1 つのルートテーブルにしか関連付けることができません。
セキュリティグループ	あるインスタンスに対して許可されている、名前が付けられた一連の受信方向のネットワーク接続。
サブネット	EC2 インスタンスをアタッチできる VPC の IP アドレス範囲の一部分。セキュリティと運用上の必要に応じて、サブネットを作成し、インスタンスをグループ分けできます。
Virtual Private Cloud (VPC)	定義した仮想ネットワーク内で AWS リソースを起動できる、AWS クラウドの論理的に隔離されたセクションをプロビジョニングする Web サービス。
Auto Scaling	ユーザー定義のポリシー、スケジュール、ヘルスチェックに基づいて Amazon EC2 インスタンスを自動的に起動または終了するウェブサービス。
クラウドの形成	関連する AWS リソースを 1 つの単位として一緒に作成および削除するテンプレートを書き込んだり変更したりするサービス。

NetScaler BLX が AWS でどのように動作するか

NetScaler BLX は、Linux システムでネイティブに動作する軽量のソフトウェアパッケージです。BLX は、BLX がサポートし、AWS marketplace で入手できる任意の Linux AMI にインストールできます。サポートされている Linux ディストリビューションの詳細については、「[サポートされているLinux ディストリビューション](#)」を参照してください。

BLX は AWS VPC 内の EC2 Linux インスタンス上で Linux プロセスとして実行されます。Linux AMI インスタンスには、最低 2 つの仮想 CPU と 2 GB のメモリが必要です。AWS VPC 内で起動された EC2 インスタンスは、複数のインターフェイスを持つことも、インターフェイスごとに複数の IP アドレスを持つこともできます。各 BLX インスタンスには、少なくとも 3 つの IP サブネットが必要です。

- 管理サブネット (NSIP)
- クライアント側サブネット (VIP)
- バックエンド向けサブネット (SNIP)

注:

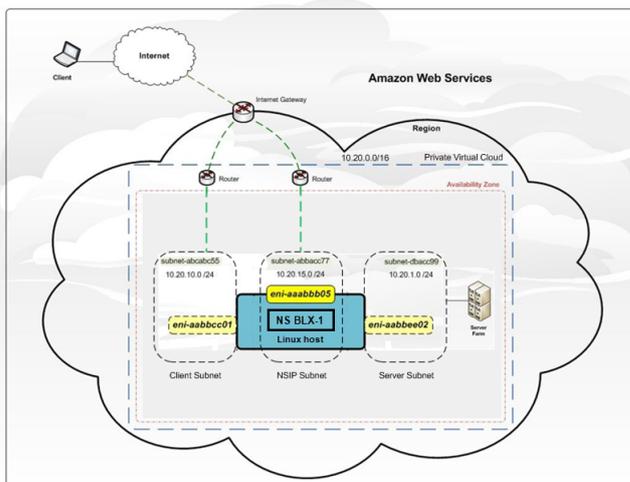
AWS での標準 BLX 展開には、3 つのネットワークインターフェイスをお勧めします。

AWS は現在、AWS VPC 内で実行されているインスタンスのみにマルチ IP 機能をサポートしています。VPC 内の BLX インスタンスを使用して、EC2 インスタンスで実行されているサーバーの負荷を分散できます。Amazon VPC を使用すれば、独自の IP アドレス範囲、サブネット、ルートテーブル、ネットワークゲートウェイなどを含めて、仮想ネットワーク環境を作成および管理できます。

注:

デフォルトでは、各 AWS アカウントの AWS リージョンごとに最大 5 つの VPC インスタンスを作成できます。Amazon のリクエストフォームを送信することで、より高い VPC 制限をリクエストできます。

次の図は、Linux AMI に BLX が展開された AWS VPC の簡単なトポロジを示しています。



AWS VPC は、以下の要素で構成されています。

- VPC の送受信トラフィックをルーティングする単一のインターネットゲートウェイ
- インターネットゲートウェイとインターネット間のネットワーク接続
- 管理、クライアント、サーバー用にそれぞれ 1 つずつ、合計 3 つのサブネット
- インターネットゲートウェイと 2 つのサブネット (管理とクライアント) 間のネットワーク接続
- 各サブネットに 3 つの ENI が接続されている Linux インスタンスにインストールされたスタンドアロン BLX インスタンス

前提条件

AWS でインスタンスを作成する前に、次の点を確認してください。

- EC2 インスタンスが [BLX システム要件を満たしていることを確認](#) します。
- パフォーマンスを向上させるには、m5.xlarge 以上のインスタンスタイプを作成することをお勧めします。
- NSIP、VIP、SNIP を構成するには、3 つの IP アドレスが必要です。

注:

VIP および SNIP として設定された IP アドレスには、パブリック IP アドレスが関連付けられている必要があります。

- AWS 仮想プライベートクラウド (VPC) で Linux AMI を起動するには、AWS アカウントが必要です。AWS アカウントは aws.amazon.com で無料で作成できます。
- ユーザーの AWS サービスとリソースへのアクセスを安全に制御するには、AWS ID およびアクセス管理 (IAM) ユーザーアカウントが必要です。IAM ユーザーアカウントの作成方法の詳細については、「[IAM ユーザーの作成 \(コンソール\)](#)」を参照してください。
- ターミナルプログラムから AWS マネジメントコンソールのすべての機能を使用できます。詳細については、[AWS CLI ユーザーガイドを参照](#) してください。また、ネットワークインターフェイスの種類を SR-IOV に変更するには、AWS CLI も必要です。
- Elastic Network Adapter (ENA) ドライバー対応のインスタンスタイプ (M5、C5 インスタンスなど) の場合、ファームウェアのバージョンは 13.0 以降である必要があります。

制限事項と使用ガイドライン

NetScaler BLX インスタンスを AWS に展開する場合、以下の制限と使用ガイドラインが適用されます。

- データトラフィックと管理トラフィック ENI は異なるサブネットにある必要があります。
- 管理 ENI には NSIP アドレスのみが必要です。
- NSIP に EIP を割り当てる代わりに NAT インスタンスをセキュリティに使用する場合は、適切な VPC レベルのルーティング変更が必要です。VPC レベルのルーティング変更手順については、「[シナリオ 2: パブリックサブネットとプライベートサブネットを持つ VPC](#)」を参照してください。

- 1 つの ENI に複数の IP アドレスを割り当てることができます。ENI ごとの IP アドレスの最大数は EC2 インスタンスタイプによって決まります。[Elastic Network Interfaces](#)の「インスタンスタイプごとのネットワークインターフェイスごとの IP アドレス」セクションを参照してください。

注:

IP アドレスを ENI に割り当てる前に、AWS で割り当てる必要があります。詳細については、「[Elastic ネットワークインターフェイス](#)」を参照してください。

- AWS の制限により、次の機能はサポートされていません。
 - Gratuitous ARP (GARP)
 - L2 モード
 - タグ付き VLAN
 - 動的ルーティング
 - 仮想 MAC
- RNAT が機能するには、送信元/送信先チェックが無効になっていることを確認してください。詳細については、[ElasticNetwork Interfaces](#) の「送信元/送信先チェックの変更」を参照してください。

スタンドアロンの **NetScaler BLX** インスタンスを **AWS** に展開します

August 30, 2024

このトピックでは、AWS に EC2 Linux インスタンスを作成し、その Linux インスタンスに BLX をインストールする手順について説明します。

展開を開始する前に、以下のトピックをお読みください。

- [前提条件](#)
- [制限事項と使用上のガイドライン](#)

展開手順

次の手順を実行します:

1. EC2 Linux インスタンスを作成する
2. NetScaler BLX をダウンロード
3. NetScaler BLX のインストール
4. BLX を専用モードで設定
5. NetScaler BLX を起動する
6. NetScaler BLX にアクセス
7. ライセンス

EC2 Linux インスタンスを作成する

AWS ウェブコンソールを使用して AWS に EC2 Linux インスタンスを作成するには、次のステップを実行します。

1. キーペアを作成する

Amazon EC2 はキーペアを使用してログイン情報を暗号化および復号化します。インスタンスにログオンするには、キーペアを作成し、インスタンスを起動するときにキーペアの名前を指定し、インスタンスに接続するときにプライベートキーを指定する必要があります。

AWS Launch Instance ウィザードを使用してインスタンスを確認し、起動すると、既存のキーペアを使用するか、新しいキーペアを作成するように求められます。キーペアの作成方法の詳細については、「[Amazon EC2 キーペア](#)」を参照してください。

2. VPC を作成する

NetScaler VPC インスタンスは AWS VPC 内で展開されます。VPC では、AWS アカウント専用の仮想ネットワークを定義できます。AWS VPC の詳細については、「[Amazon VPC の使用開始](#)」を参照してください。

Linux インスタンス用の VPC を作成する際は、次の点に注意してください。

- AWS アベイラビリティゾーンに AWS VPC を作成するには、「単一パブリックサブネットのみを使用する VPC」オプションを使用します。
- Citrix では、少なくとも 3 つのサブネットを作成することをお勧めします。すべてのサブネットは、同じアベイラビリティゾーンに存在する必要があります。
 - 管理トラフィック用の 1 つのサブネット。このサブネットに管理 IP (NSIP) を配置します。デフォルトでは、エラスティックネットワークインターフェース (ENI) eth0 が管理 IP に使用されます。
 - クライアントアクセス (ユーザーから NetScaler BLX) トラフィック用の 1 つまたは複数のサブネット。クライアントはこのサブネットを介して、NetScaler 負荷分散仮想サーバーに割り当てられた 1 つ以上の仮想 IP (VIP) アドレスに接続します。
 - サーバーアクセス (BLX からサーバーへ) トラフィック用の 1 つ以上のサブネット。サーバーはこのサブネットを介して、BLX が所有するサブネット IP (SNIP) アドレスに接続します。

3. サブネットの追加

VPC ウィザードを使用する場合、サブネットは 1 つだけ作成されます。要件によっては、さらにサブネットを作成したい場合があります。追加のサブネットを作成する方法の詳細については、「[VPC へのサブネットの追加](#)」を参照してください。

4. セキュリティグループとセキュリティルールの作成

受信トラフィックと送信トラフィックを制御するには、セキュリティグループを作成し、そのグループに規則を追加します。グループを作成してルールを追加する方法の詳細については、「[VPC のセキュリティグループ](#)」を参照してください。

EC2 ウィザードは、AWS Marketplace が生成する Linux インスタンスのデフォルトのセキュリティグループを提供します。ただし、要件に応じてさらにセキュリティグループを作成できます。

セキュリティグループの SSH、HTTP、HTTPS アクセス用に次のポートを開く必要があります。

アクセスタイプ	ポート番号
SSH	22
HTTP	80
HTTPS	443

5. ルートテーブルの追加

ルートテーブルには、ネットワークトラフィックの送信先を決定するために使用されるルートと呼ばれる一連のルールが含まれています。VPC の各サブネットはルートテーブルに関連付ける必要があります。ルートテーブルの作成方法について詳しくは、「[ルートテーブル](#)」を参照してください。

6. インターネットゲートウェイを作成する

インターネットゲートウェイには 2 つの目的があります。1 つは、インターネットでルーティング可能なトラフィックのターゲットを VPC ルートテーブルに指定すること、パブリック IPv4 アドレスが割り当てられたインスタンスに対してネットワークアドレス変換 (NAT) を行うことです。

インターネットトラフィックに対して、インターネットゲートウェイを作成します。インターネットゲートウェイの作成方法の詳細については、「[インターネットゲートウェイをアタッチする](#)」を参照してください。

7. AWS EC2 サービスを使用して Linux インスタンスを作成する

- a) AWS ダッシュボードから、[コンピューティング] > [EC2] > [インスタンスの起動] > [AWS marketplace] に移動します。

[Launch Instance] をクリックする前に、Launch Instance の下に表示されるメモを確認して、リージョンが正しいことを確認してください。

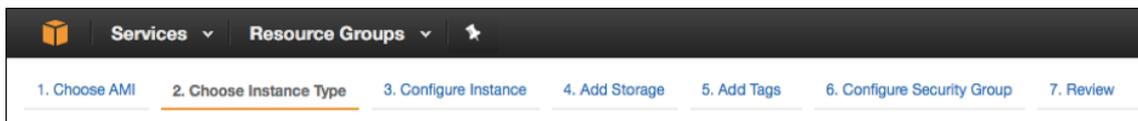


- b) 「AWS Marketplace 検索」バーで、「Linux」というキーワードで検索します。
c) 展開する AMI を選択し、「選択」をクリックします。

Launch Instance ウィザードが起動します。ウィザードに従って、インスタンスを作成します。このウィザードでは、次のことを求められます。

- インスタンスの種類の選択

- インスタンスの構成
- ストレージの追加
- タグの追加
- セキュリティグループの構成
- 確認



8. ネットワークインターフェースをさらに作成してアタッチする

VIP と SNIP 用に 2 つのネットワークインターフェイスを作成します。ネットワークインターフェイスをさらに作成する方法の詳細については、「[ネットワークインターフェイスの作成](#)」セクションを参照してください。

ネットワークインターフェイスを作成したら、それらを Linux インスタンスにアタッチする必要があります。インターフェイスをアタッチする前に、Linux インスタンスをシャットダウンし、インターフェイスをアタッチして、インスタンスをパワーオンします。ネットワークインターフェイスをアタッチする方法の詳細については、「[インスタンス起動時にネットワークインターフェイスをアタッチする](#)」セクションを参照してください。

9. Elastic IP アドレスの割り当てと関連付け

EC2 インスタンスにパブリック IP アドレスを割り当てた場合、そのアドレスはインスタンスが停止されるまで割り当てられたままになります。その後、アドレスはプールに解放されます。インスタンスを再起動すると、新しいパブリック IP アドレスが割り当てられます。

対照的に、エラスティック IP (EIP) アドレスの場合は、インスタンスから割り当てが解除されるまで割り当ての状態が維持されます。

管理 NIC に Elastic IP を割り当てて関連付けるには、「[Elastic IP アドレスの割り当て](#)」を参照してください。

これらのステップにより、AWS で Linux インスタンスを作成する手順は完了です。インスタンスの準備が完了するまで数分かかる場合があります。インスタンスがステータスチェックに合格したことを確認します。この情報は、** インスタンスページのステータスチェック列で確認できます **。

10. Linux インスタンスに接続する

Linux インスタンスを作成したら、AWS マネジメントコンソールからインスタンスに接続できます。

- a) Linux インスタンスを選択して [接続] をクリックします。
- b) 「インスタンスへの接続」 ページの指示に従ってください。

NetScaler BLX をダウンロード

Linux AMI に BLX パッケージをダウンロードしてください。詳しくは、「[NetScaler BLX のダウンロード](#)」を参照してください。

NetScaler BLX のインストール

- Debian ベースの Linux ホストに BLX をインストールするには、「[Debian ベースの Linux ホストへの NetScaler BLX のインストール](#)」を参照してください。
- RPM ベースの Linux ホストに BLX をインストールするには、「[RPM ベースの Linux ホストへの NetScaler BLX のインストール](#)」を参照してください。

BLX を専用モードで設定

BLX をインストールしたら、設定ファイルを編集して BLX を専用モードで起動します。詳しくは、「[専用モードでの NetScaler BLX の構成](#)」を参照してください。

NetScaler BLX を起動する

blx.conf ファイルを編集したら、BLX を起動します。詳しくは、「[NetScaler BLX の起動](#)」を参照してください。

NetScaler BLX にアクセス

BLX には、次のいずれかの方法でアクセスできます。

- NetScaler CLI
- NetScaler GUI
- NetScaler NITRO REST API

詳しくは、「[NetScaler BLX へのアクセス](#)」を参照してください。

ライセンス

NetScaler は、組織の要件を満たす幅広い製品エディションとライセンスモデルを提供しています。詳しくは、「[NetScaler BLX ライセンス](#)」を参照してください。

NetScaler BLX と GSLB を AWS に展開

August 30, 2024

NetScaler BLX は NetScaler のソフトウェアフォームファクターであり、他のフォームファクターと同じ機能を提供します。Linux ホスト上でユーザースペースアプリケーションとして実行されます。

GSLB for BLX を AWS にセットアップするには、NetScaler が属する VPC の外部にあるサーバー（異なるアベイラビリティリージョンの別の VPC 内やオンプレミスのデータセンターなど）にトラフィックを負荷分散するように NetScaler を構成する必要があります。



ドメインベースのサービスの概要

NetScaler GSLB は、クラウドロードバランサーのドメインベースサービス (DBS) をサポートしているため、動的なクラウドサービスを自動的に検出できます。この構成により、NetScaler はアクティブ-アクティブ環境でグローバルサーバー負荷分散を実装できます。DBS では、DNS 検出を通じて AWS 環境のバックエンドリソースをスケールリングできます。

DBS と ELB

GSLB DBS は、ユーザー Elastic Load Balancer (ELB) の FQDN を使用して、AWS 内で作成および削除されるバックエンドサーバーを含むように GSLB サービスグループを動的に更新します。AWS のバックエンドサーバーまたはインスタンスは、ネットワーク需要または CPU 使用率に基づいてスケールするように設定できます。この機能を構成するには、NetScaler を ELB にポイントして、AWS 内でインスタンスが作成および削除されるたびに NetScaler を手動で更新しなくても、AWS 内のさまざまなサーバーに動的にルーティングできます。GSLB サービスグループの NetScaler DBS 機能は、DNS 対応サービス検出を使用して、Autoscale e グループで識別される DBS 名前空間のメンバーサービスリソースを決定します。

前提条件

- 2つのスタンドアロン BLX インスタンスを異なる AWS VPC に展開します。展開について詳しくは、「[スタンドアロン NetScaler BLX インスタンスを AWS に展開する](#)」を参照してください。

注:

blx.confファイルの cli-cmds セクションで次のコマンドを指定していることを確認します。

```
- add ns ip <VIP address> <netmask> -type VIP
- add ns ip <SNIP address> <netmask> -type SNIP -mgmtAccess
  ENABLED
```

- GSLB 機能をサポートする NetScaler ライセンスが必要です。詳しくは、「[ライセンス](#)」を参照してください。
- AWS で 2 つの Linux インスタンスが使用可能であることを確認します。

注:

パフォーマンスを向上させるには、m5.xlarge 以上のインスタンスタイプを作成することをお勧めします。

- NSIP、VIP、SNIP として構成するには、3 つの IP アドレスが必要です。

注:

VIP および SNIP として設定された IP アドレスには、パブリック IP アドレスが関連付けられている必要があります。

- セキュリティグループの次のポートを開く必要があります。
 - 53/UDP
 - 22/TCP
 - 3008/TCP
 - 3009/TCP

blx.conf ファイルのサンプル

```
1 blx-system-config
2 {
3
4     core-dumps: yes
5     syslog: yes
6     ipaddress: 10.0.12.245/20
7     interfaces: eth1
8 }
9
10 static-routes
11 {
12
13     default 10.0.0.1
14 }
15
```

```
16 cli-cmds
17 {
18
19     add ns ip 11.0.12.245/20 -type VIP`
20     add ns ip 12.0.12.245/20 -type SNIP -mgmtAccess ENABLED`
21 }
```

GSLB の設定

GSLB で構成された NetScaler は、WAN の障害点から保護することで、災害復旧を実現し、アプリケーションの継続的な可用性を確保します。GSLB は、クライアントのリクエストを最も近いまたは最もパフォーマンスの高いデータセンター、または障害が発生した場合は存続しているデータセンターに転送することで、データセンター間の負荷を分散します。

GSLB の展開タイプと構成の詳細については、[GSLBドキュメント](#)を参照してください。

よくある質問

September 30, 2024

NetScaler BLX とは何ですか？

NetScaler BLX は、Linux ホスト上でネイティブ アプリケーションとして実行される NetScaler のベアメタル ソフトウェア バージョンです。詳細については、「[NetScaler BLX について](#)」を参照してください。

NetScaler BLX が必要な理由は何ですか？

NetScaler BLX は NetScaler のベアメタル バージョンであり、仮想マシンのオーバーヘッドがなくシンプルであるため、パフォーマンスが向上します。また、NetScaler BLX を任意のサーバー ハードウェアで実行することもできます。詳細については、「[NetScaler BLX を選ぶ理由](#)」を参照してください。

NetScaler BLX、CPX、VPX の違いは何ですか？

NetScaler BLX、CPX、VPX は、NetScaler のソフトウェア フォーム ファクターです。

NetScaler VPX は仮想アプライアンスであり、サーバーにインストールされたハイパーバイザー上で実行する必要があります。

NetScaler CPX は、コンテナ内で実行する必要がある NetScaler のコンテナ化されたバージョンです。

NetScaler BLX は、Linux システム上でネイティブに実行されるソフトウェア パッケージです。

NetScaler BLX はいつ使用すればよいですか？

NetScaler VPX、CPX、NetScaler BLX は、ハイブリッド マルチクラウドへの移行をサポートする、業界で最も包括的なソフトウェア中心の ADC ラインナップです。次の表は、違いと使用例についてのガイドラインを示しています。

製品	ユースケース	特徴
NetScaler VPX (ハイパーバイザー上で実行)	ハードウェア インフラストラクチャの仮想化、共通インフラストラクチャ上でのワークロードの統合	ハードウェアと OS に依存せず、完全な分離とマルチテナントのサポート
NetScaler CPX (コンテナ内で実行)	DevOps、マイクロサービス、自動ステージング、テスト、デプロイメント、東西トラフィック	軽量、小型、API ゲートウェイ機能、マイクロサービス中心、認証
NetScaler BLX (ベアメタル サーバー上で実行)	高いトラフィック負荷、ミッションクリティカルなアプリケーション、レイテンシに敏感なワークロード、南北トラフィック	ネイティブ Linux ソフトウェアパッケージで VM オーバーヘッドなし

ハイパーバイザーやコンテナがない場合、どのような違いが生じますか？

ハイパーバイザー変換レイヤーやコンテナがないため、NetScaler BLX ソフトウェアは基盤となるハードウェアをより細かく制御でき、パフォーマンスが向上します。また、ハイパーバイザー ソフトウェアには追加コストはかかりません。

NetScaler BLX はどのサーバー ハードウェアでも実行できますか？

はい、NetScaler BLX はどのサーバー ハードウェアでも実行できます。ただし、より高いパフォーマンスを得るには、NetScaler BLX でサポートされている DPDK 互換 NIC を使用することをお勧めします。ハードウェア要件の詳細については、「[Linux ホストのハードウェア要件](#)」を参照してください。

NetScaler BLX を Linux サーバーに導入するにはどうすればよいですか？

NetScaler BLX は、次のいずれかのパッケージ配布をサポートする任意の Linux サーバーに展開できます。

- `.rpm` (RPM ベースのパッケージ)
- `.deb` (Debian ベースのパッケージ)

NetScaler BLX の展開の詳細については、「[NetScaler BLX の展開](#)」を参照してください。

NetScaler BLX ソフトウェアの展開を自動化できますか？

はい、NetScaler BLX を展開するには、`.rpm` または `.deb` パッケージをサポートする任意のソフトウェア展開ツールを使用できます。

たとえば、Terraform を使用して NetScaler BLX をデプロイできます。詳細については、「[Terraform を使用した NetScaler BLX の展開](#)」を参照してください。

NetScaler BLX を **Linux OS** 搭載サーバーにインストールした場合、同じサーバーに他の標準 **Linux** パッケージやアプリケーションをインストールできますか？

はい、他の標準 Linux パッケージまたはアプリケーションを NetScaler BLX と並行して実行できます。

NetScaler BLX ライセンスを購入するにはどうすればよいですか？

デフォルトでは、NetScaler BLX には Express ライセンスが付属しています。NetScaler BLX は無料でお試しいただけます。

製品に満足したら、サブスクリプションベースのローカル ライセンスまたは NetScaler プール容量ライセンスにアップグレードできます。

NetScaler BLX ライセンスの詳細については、「[NetScaler BLX ライセンス](#)」を参照してください。

現在の **NetScaler VPX** ライセンスを **NetScaler BLX** に使用できますか？

はい、現在の VPX ライセンスを NetScaler BLX に使用できます。詳細については、「[NetScaler BLX ライセンス](#)」を参照してください。

NetScaler BLX をワンアーム モードとツーアーム モードで展開できますか？

はい、NetScaler BLX はワンアーム モードまたはツーアーム モードのいずれかで展開できます。詳細については、「[物理展開モード](#)」を参照してください。

NetScaler BLX はどのネットワーク スタックを使用しますか？

専用モードの NetScaler BLX は独自のネットワーク スタックを使用します。NetScaler BLX アーキテクチャの詳細については、「[一般的なアーキテクチャ](#)」を参照してください。

NetScaler BLX は高可用性をサポートしていますか？

NetScaler BLX では、高可用性は専用モードでのみサポートされます。高可用性設定の詳細については、「[高可用性](#)」を参照してください。

NetScaler BLX と **NetScaler VPX** または **CPX** の間に高可用性ペアを設定できますか？

いいえ、NetScaler BLX と VPX または CPX の間に高可用性ペアを設定することはできません。

DPDK を搭載した仮想化 **Linux** マシンで **NetScaler BLX** を実行できますか？

はい、仮想化された Linux マシン上で NetScaler BLX を実行できます。

NetScaler BLX を **ARM** プラットフォームで実行できますか？

いいえ、NetScaler BLX は Intel または AMD x86-64 (64 ビット) Linux プラットフォームでのみサポートされません。

NetScaler BLX の管理 IP アドレスは何ですか？

NSIP アドレスは、NetScaler BLX の管理 IP アドレスです。これは、`blx.conf` の `ipaddress` パラメータを使用して設定されます。`ipaddress` パラメータで NSIP アドレスを指定せずに NetScaler BLX 管理対象ホストを構成した場合、Linux ホスト IP が NSIP アドレスとして NetScaler BLX に自動的に追加されます。

NetScaler BLX でメモリ使用率が常に高いのはなぜですか？

メモリ使用率は、NetScaler BLX の現在のメモリ使用量と、NetScaler BLX がオペレーティング システム (OS) から割り当てるメモリの合計量を比較して決定されます。NetScaler BLX は、起動と基本操作に必要な最小限のメモリを OS から割り当てることから開始します。当初、NetScaler BLX は操作に必要な最小限のメモリを割り当てるため、メモリ消費率は高くなります。最終的には、NetScaler BLX 操作のスケーリングに伴ってリソースの必要性が高まり、NetScaler BLX は OS から追加のメモリを段階的に割り当て、メモリ使用量も高くなり、結果としてメモリ使用率が高くなります。

したがって、NetScaler BLX の動的メモリ割り当て戦略により、NetScaler BLX は通常、使用可能なメモリの不足ではなく、リソースの効率的な使用を反映して、高いメモリ使用率を示します。

Troubleshooting

May 1, 2024

blx.conf ファイル内の **ipaddress (NSIP)** を変更しましたが、**NetScaler BLX** は引き続き古い **NSIP** アドレスを使用しています

専用モードの NetScaler BLX の管理 IP アドレスは、次のいずれかの方法で構成しない限り、常に **blx.conf** ファイルの **ipaddress** パラメーターに設定された IP アドレスです。

- **NetScaler BLX CLI:** NetScaler BLX CLI で `set ns config` コマンドを実行し、管理 IP アドレスを変更します。構成の変更は、NetScaler BLX の保存済み構成ファイル (`/nsconfig/ns.conf`) に保存されます。
- **NetScaler BLX GUI:** NetScaler BLX GUI の構成ユーティリティ画面で、右上隅にある歯車アイコンをクリックし、**NSIP** アドレスペインをクリックして管理 **IP** アドレスを変更します。

行った構成の変更は、NetScaler BLX で保存された構成ファイル (`/nsconfig/ns.conf`) に保存されます。

`ns.conf` ファイルでの変更は、常に `blx.conf` ファイルよりも優先されます。

systemctl start blx コマンドを使用して **NetScaler BLX** を起動しましたが、起動に失敗します

- NetScaler BLX ブートログファイル (`/var/log/blx-boot.log`) で、NetScaler BLX 構成ファイル (`/etc/blx/blx.conf`) の解析エラーに関連するログを探してください。
- SYSLOG ファイル (`/var/log/messages`) でクラッシュ関連のログまたはエラーログを探します。

NetScaler BLX には **DPDK** ポートが搭載されていません

- **blx.conf** ファイルの **interfaces** パラメーターで指定されているポートが、NetScaler BLX がサポートする DPDK ポートであることを確認してください。NetScaler BLX でサポートされる DPDK ポートについては、「[Linux ホストのハードウェア要件](#)」を参照してください。

NetScaler BLX の制限と使用ガイドライン

September 30, 2024

以下の制限事項と使用ガイドラインは NetScaler BLX に関連しています。

高可用性

- 高可用性は、Amazon Web Services (AWS) や Oracle Cloud Infrastructure (OCI) などのパブリッククラウドプラットフォームではサポートされていません。
- `nsinternal` ユーザー ログインが無効になっている場合、高可用性はサポートされません。
- 高可用性は専用モードでのみサポートされます。

NetScaler BLX クラスタ

- INC モードはサポートされていません。
- CLAG ベースのトラフィック分散はサポートされていません。

NetScaler ゲートウェイ

- MAC および Linux SSO VPN クライアントはサポートされていません。
- RDP プロキシ機能はサポートされていません。

LA および LACP チャンネル

- LA/LACP チャンネルは共有モードではサポートされません。
- LA/LACP チャンネルは、専用 NIC インターフェイスまたは DPDK NIC インターフェイス間でのみサポートされます。
- LA/LACP チャンネルは、`blx1` および `ns1` 仮想インターフェイスではサポートされていません。

SNMP の

- SNMP は専用モードの BLX でのみサポートされます。

ウェブ アプリケーション ファイアウォール (WAF)

- Web アプリケーション ファイアウォール (WAF) は、専用モードの NetScaler BLX でのみサポートされます。
- Web アプリケーション ファイアウォール (WAF) が有効になっている場合、BLX ゲートウェイにアクセスできません。

DPDK ポートを備えた NetScaler BLX

- Linux ホストが Intel(R) Xeon(R) CPU E5-2690 v4 @ 2.60 GHz や CPU E5504 @ 2.00 GHz などの古い CPU モデルで実行されている場合、DPDK ポートを備えた BLX の起動に失敗する可能性があります。
- BLX の実行中に DPDK モジュールにバインドされた NIC ポートのバインドを解除すると、Linux ホストがクラッシュする可能性があります。
- DPDK ポートを備えた BLX は、DPDK ポートを備えていない BLX よりも再起動に少し時間がかかります。
- すべての DPDK バインド Linux ポートは自動的に BLX 専用となり、他の DPDK Linux アプリケーションには使用できません。
- BLX でサポートされている VMXNET3 DPDK ポートの場合、ワーカー プロセスの数を 2 の累乗 (2^n) で指定する必要があります。たとえば、1、2、4、8 などです。
- BLX は、DPDK ポートに対してのみトランク モードまたは VLAN タグ付けをサポートします。

Mellanox ポート

- BLX は一度に 1 種類の DPDK ポートのみをサポートします。たとえば、すべて Mellanox ポート、またはすべて Intel ポートのいずれかです。
- BLX は Mellanox ポート用の MLX5 DPDK ドライバーのみをサポートします。
- MLX5 DPDK ドライバーとその制限の詳細については、公式の [MLX5 DPDK ドキュメント](#) を参照してください。
- Mellanox NIC とその制限事項の詳細については、公式の [Mellanox ドキュメント](#) を参照してください。

その他の制限とガイドライン

- `set ns hostname` コマンドを使用して BLX のホスト名を設定すると、Linux ホストのホスト名も変更されます。
- BLX 管理対象ホスト機能が設定された BLX を再起動すると、Linux ホストへのアクティブな SSH セッションがすべて閉じられます。接続を復元するには、ホストへの接続を再試行する必要があります。
- 専用モードでは、`blx.conf` ファイルで指定された管理 HTTP または HTTPS ポート (`mgmt-http-port` または `mgmt-https-port`) は無視されます。デフォルトでは、ポート番号 80 と 443 は HTTP および HTTPS 管理アクセス専用です。専用モードで BLX のこれらのポートを変更するには、次の NetScaler CLI コマンドを使用する必要があります。

```
set ns param (-mgmthttpport <value> | -mgmthttpsport <value>)
```

例: 次のコマンドは、管理 HTTP ポートを 2080 に変更します。

```
set ns param -mgmthttpport 2080
```

- Linux ホストでファイアウォールが有効になっている場合は、BLX 管理ポートと SYSLOG ポートの例外を追加する必要がある場合があります。
- BLX の起動には最大 45 秒かかる場合があります。
- BLX 設定は `/nsconfig/ns.conf` ファイルに保存されます。構成をセッション間で利用できるようにするには、構成を変更するたびに構成を保存する必要があります。

– **NetScaler CLI** を使用して実行中の構成を表示するには

コマンドプロンプトで次のように入力します。

```
show ns runningConfig
```

– **NetScaler CLI** を使用して設定を保存するには

コマンドプロンプトで次のように入力します。

```
save ns config
```

- `/nsconfig/ns.conf` ファイル内の BLX 構成は、`/etc/blx/blx.conf` ファイル内の構成よりも優先されます。
- 割り当てられたメモリがワーカー プロセスごとに 1 GB 未満の場合、BLX は起動しません。
- BLX をインストールすると、Linux ホスト上の `ip_forward` パラメータが 1 に設定されます。
- BLX をアンインストールした後、設定ファイル (`blx.conf`) は保持され、`blx.conf.rpmsave` としてバックアップされます。このバックアップ構成ファイルを同じ Linux ホストに新しくインストールされた BLX に適用するには、ファイルの名前を手動で `blx.conf` に変更する必要があります。
- BLX ではパケットドロップ関連の問題が発生する可能性があるため、次の Ubuntu バージョンで BLX を実行することはお勧めしません。

Ubuntu version 16.04.5 with kernel version 4.4.0-131-generic

- BLX は最大 9 個の NIC ポート (DPDK NIC ポート、非 DPDK NIC ポート、またはその両方の組み合わせ) をサポートします。
- 次の条件が満たされる場合、BLX は起動しないか、正常に機能しない可能性があります。
 - SELinux ポリシーが Linux ホストで有効になっています。SELinux は、`systemd` プロセスが一部の BLX システム ファイルを実行するのを防ぎます。
回避策: Linux ホストで SELinux を無効にします。

注記:

NetScaler BLX 14.1-17.x 以降では、Red Hat ベースの Linux ホストに BLX をインストールすると、Linux ホストで SELinux モジュールが使用可能な場合は SELinux ポリシーが適用されます。このポリ


```
Opswat) for Mac\" type=\"MAC-EPA-ENGINE\"\\n\\t\\t\\tversion
=\"1.3.5.7\" path=\"/epa/scripts/mac/MacLibs.zip\"
opswatVersion=\"4.3.2138.0\"\\n\\t\\t/>\\n\\n\\t\\t<plugin\\n\\t\\t\\
tname=\"Netscaler Gateway RfWeb GUI\" type=\"RFWEB-GUI\"\\n
\\t\\t\\tversion=\"23.8.1.11\" path=\"/logon/logonPoint/\"\\n\\
t\\t/>\\n\\t</repository>\\n</repositories>\\n\" }
```

3. 「」

レスポンスポリシー pluginlist_respond_pol “HTTP.REQ.URL.CONTAINS(“pluginlist.xml”)”
pluginlist_respond を追加します

“

```
1 bind vpn vserver <VSERVER_NAME> -policy pluginlist_respond_pol
-priority 1 -gotoPriorityExpression END -type AAA_REQUEST
```

5. BLX を非ルート nsroot モードで起動すると、ルート nsroot ユーザー モードに戻ることはできません。そのためには、BLX を再インストールする必要があります。showtechsupport コマンドを実行するには、ホストに root としてログインし、showtechsupport.pl スクリプトを実行する必要があります。

NetScaler BLX でサポートされていない NetScaler 機能

- 管理パーティション
- コンテンツの最適化
- ハードウェア SSL オフロード
- 中間システム間 (IS-IS) ルーティング プロトコル
- IPSec
- ジャンボフレーム
- 高精度時間プロトコル (PTP)
- サービス品質 (QoS)
- ルーティング情報プロトコル (RIP)
- ルーティング情報プロトコル次世代 (RIPng)
- URL フィルタリング



© 2024 Cloud Software Group, Inc. All rights reserved. Cloud Software Group, the Cloud Software Group logo, and other marks appearing herein are property of Cloud Software Group, Inc. and/or one or more of its subsidiaries, and may be registered with the U.S. Patent and Trademark Office and in other countries. All other marks are the property of their respective owner(s).
